Protagonist vs Antagonist PROVANT: Narrative Generation as Counter Planning

Socially Interactive Agents Track

Julie Porteous
RMIT University, Australia
julie.porteous@rmit.edu.au

Alan Lindsay
University of Huddersfield, UK
A.Lindsay@hud.ac.uk

ABSTRACT

Our motivation in this work is to develop a narrative generation mechanism for Interactive Storytelling that removes some of the authoring burden that is inherent to plan-based approaches. We focus on the class of narratives that dominate in Hollywood movies, television serial dramas and situation comedies. These narratives revolve around a central Protagonist in pursuit of a goal and who faces a series of obstructions placed in their way by an Antagonist and which they must overcome in order to reach their goal.

We cast this problem as a non-cooperative multi-agent planning problem, in other words counter planning. We build on recent techniques in goal recognition and landmark identification to develop a novel plan-based narrative generation mechanism. A key opportunity that goal recognition provides is to reason explicitly with partially observed action sequences, reflecting the reasoning process of the antagonist. Thus the antagonist can only act to obstruct if it is reasonable (to the viewer) that they have guessed the protagonist's intentions. Starting from the believed goal, the narrative generator can reason about the protagonist's plan and what *must* be done to achieve it i.e., the plan landmarks [8] and use these to automatically identify suitable points of obstruction. In the paper we detail the approach and illustrate it with a worked example. We report the results of an experimental evaluation and user study in a number of representative narrative domains. The experimental results show that we can construct narratives displaying the desired structure without the overhead of authoring narrative structuring information. Results of the user study with system generated narratives confirm that viewers can clearly recognise agent roles and narrative structure.

KEYWORDS

Interactive Narrative and Storytelling; Entertainment

ACM Reference Format:

Julie Porteous and Alan Lindsay. 2019. Protagonist vs Antagonist PROVANT: Narrative Generation as Counter Planning. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 9 pages.

1 INTRODUCTION

AI Planning has been widely used for narrative generation in Interactive Storytelling and Games [5, 16, 24, 31], largely because a plan based approach can help ensure causality (promoting story understanding), whilst also offering a powerful and flexible generation mechanism [34]. However, the creation of planning domain models has been acknowledged as a bottleneck that limits the application of AI planning [9] and for narrative applications this is further compounded by the need to author additional intermediate agent goals in order to generate narratives which display suitable structure and make sense to an audience. For example the inclusion of authorial goals by Reidl [23] or pseudo-landmarks by Porteous et al [18]. Thus in this work our motivation is to develop a narrative generation mechanism which outputs narrative variants displaying desired narrative plan structure without requiring additional authoring overhead of structuring information.

Our starting point was the observation that in many genres narratives conform to what David Bordwell described as the *Canonic Story Form* [3] something that has also been noted in research on story understanding and story grammars such as [27, 32]. Within this the story is organised around a single *Protagonist* in pursuit of some goal and it is their actions that drive the plot. The narrative starts with a phase of exposition to clearly show the motivation of the protagonist and the goal which they are striving towards. The protagonist might have allies but their goal is attained primarily through their own actions. The protagonist is blocked in the pursuit of their goal by an *Antagonist* and the narrative proceeds with a series of obstacles which the protagonist must overcome in order to reach their goal. Most frequently (but not always) the protagonist happily achieves their goal. This classical canonic story form is prevalent in Hollywood and also in television serial dramas and situation comedies. As illustration consider the movie Raiders of the Lost Ark [30] where Indiana Jones (the protagonist) is obstructed in his quest to find the Ark by the Nazis (the antagonist) who place obstructions in his way.

Thus the problem we tackle is generating narrative plans for a protagonist agent: beginning with their trait-driven pursuit of their goal followed by a series of obstructions caused by an antagonist agent, each of which forces the protagonist to find an alternative way of achieving their goal. An important aspect of the solution to this problem is that the generation mechanism should not need narrative structuring information to be included in the narrative domain model.

We observe that this is similar to the problem of planning for a non-cooperative multi-agent system that features agents who wish to prevent opponents from achieving their goals: a problem which has been referred to as counter planning [4]. Recent results in this area include the fully automated approach of Pozanco et al [21] which uses a combination of: goal recognition to infer a *seeking* agent's goal; landmark extraction to identify subgoals that

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.

Exposition
1: **(introduce-pro john)**
2: **(introduce-ant tom)**

(:init (protagonist john) (antagonist tom) (geek john)
(has-crush john ann) (hates tom john) ...)
(:goal (gone-prom john ann))

Episode 1
Pro Plan
3: **(join-band john)***
4: **(play-in-band john)**
↓ (has-instrument john) **
5: ✗(go-see-band ann)
6: (ask-to-prom john ann)
7: (drive-to-prom john ann)
8: (go-prom john ann)
Ant Plan
9: **(steal-instrument tom john)**

Episode 2
Pro Plan
10: **(wash-dishes john)**
11: **(buy-instrument john)**
12: **(go-see-band ann)**
13: **(ask-to-prom john ann)**
↓ (car-working john) **
14: ✗(drive-to-prom john ann)
15: (go-prom john ann)
Ant Plan
16: **(break-car tom)**

Episode 3 Resolution
Pro Plan
17: **(borrow-car john)**
18: **(drive-to-prom john ann)**
19: **(go-prom john ann)**

KEY:
* Goal Recognition; ** Interference; ↓ Action precondition

Figure 1: School Example: initial exposition of Protagonist goal and Antagonist motivation(actions 1-2); Initial Pro Plan generated (episode 1, actions 3-8); * Ant recognises Pro goal by action 4; ** Ant selects prec of action 5 as Interference; Ant interference plan (episode 1, action 9); protagonist replans to goal (episode 2, actions 10-15); ** Ant selects precs of action 14 to interfere and generates plan (episode 2, action 16); Pro replan to reach goal (episode 3, actions 17-19). Output narrative is chain of 3 episodes with episode 3 resolving to the goal: the sequence of actions 1, 2, 3-4, 9, 10-13, 16-19 (highlighted bold).

can be used by a *preventing* agent to block the seeking agent's goal achievement; and AI planning by the preventing agent to generate the counter plan that blocks the seeker's goal achievement. However, we are interested in generating narratives so a key difference to counter planning is that we want to generate plans where the antagonist (preventing agent) obstructs the protagonist (seeking agent) in pursuit of their goal in ways which are *recoverable* and allow the protagonist to eventually achieve their goal (e.g. Indiana gets the Ark). Consequently our narrative generation approach uses a combination of: plan generation (the protagonists initial plan); goal recognition (by the antagonist); landmark identification (by the antagonist to select obstruction points); plan generation (the antagonists obstruction plan); re-planning (or plan repair) by the protagonist to recover from antagonist obstruction. This approach is fully implemented in a prototype system called ProVant.

The paper is organised as follows: the next section gives overview and rationale to our ProVant approach. This is followed in section 3 with detailed discussion of the ProVant algorithm and a worked example as illustration (section 4). Results of a series of experiments and a user study are reported and discussed in section 5. We close with discussion of closely related work (section 6) and conclusions.

2 RATIONALE: Protagonist vs. Antagonist

A common approach for narrative generation, has been to use goal directed planning in order to generate a plot from a domain theory. This plot-based approach to narrative generation, in the tradition of [35] such as [24], has a number of strengths including the ability to generate causally related action sequences and control over narrative properties (e.g. build up and release of suspense [6]).

An area of research that has received much attention in narrative generation is in defining an appropriate search approach that extends a basic goal directed search, by encompassing the various narrative parameters. Several approaches have been made in order to capture reasonable antagonistic behaviours. As illustration, these include: the use of specially authored narrative constraints [18]; the inclusion of authorial goals [23]; and the inclusion of character goals in the intentional planning approach of [24, 33]. However, in each case it has been necessary to manage the problem of competing goals within a single goal directed system. In so doing the burden is transferred to the domain modeller, who must structure the model in such a way that forces any generated narratives to appear to simulate the agents acting towards their competing goals.

However it appears natural instead to think of this as a multi-agent problem and separate the consideration of the two agents: the protagonist versus the antagonist. This allows us to focus on the interaction of the two agents and the properties that exist between the two agents outwith of their specific domain models. For example, we can consider how the antagonist can become aware of the protagonist and their aims (i.e., through partial observation and goal recognition) and how they make choices between alternative ways to interfere with the protagonists plan.

In this section we will introduce the protagonist and antagonist agent components of our system and in section 3 we will detail the approach to bringing these agent actions together in a narrative. As illustration throughout the paper we will use a narrative example set in the domain of a senior school. This is a familiar narrative setting which has featured in many movies, TV series, serial dramas, as well as Interactive Storytelling systems (e.g. the anti-bullying system FearNot![1] and Prom Week [12]).

2.1 The Protagonist

The protagonist can be considered as a relatively simple agent: they have some goal that they wish to achieve and they make a plan that they hope will lead them to achieve it. For example, a student at senior school who wants to get another student to go to the Prom with them: they might realise that they need to make themselves more popular and hence set about making that happen (this is the narrative example shown in Figure 1). If at some point their plan is interfered with (e.g. by an enemy) then they simply formulate a new plan to their goal and try that instead.

On their own, the actions of the protagonist in pursuit of their goal does not make an interesting narrative. For example, Bordwell notes that "the classical Hollywood film presents individuals who struggle to ... attain specific goals. In the course of this struggle the character enters into conflict with others ..." [3] (pg 157) and McKee observes that "... meaningful change in the life situation of a character is ... achieved through conflict" [14] (pg 34). Consequently the structure of the "classical Hollywood" narrative is as follows: begin with initial exposition of the protagonist goal and the antagonist motivation; proceed with the protagonists attainment of their goal blocked by antagonist interference, so the narrative consists of a series of chains of antagonist interference followed by protagonist actions to recover from the interference; these chains of obstruction/interference and recovery usually (but not necessarily) lead to the protagonist achieving their goal. It is this structure which we use as the basis for the generation mechanism in ProVant.

2.2 The Antagonist

For our target class of narratives, the role of antagonist in the narrative is to act as the force of opposition to interfere and obstruct the protagonist in achieving their goals. From the antagonist agent perspective this task can be split into two parts: observing the protagonist's behaviour in order to identify their intentions and recognise the goal that they are working towards; and considering the possible ways in which the antagonist can attempt to interfere with these intentions to stop them from achieving their goal.

For example, in the senior school narrative domain, if the protagonist student has a goal of securing a date to the prom they could start by setting in motion a series of actions to make themselves more popular (e.g. join a band, play a gig, gain social status and so on). The antagonist notices what the protagonist is doing and tries to work out their goal (intention recognition). The antagonist then reasons about what they could do to stop them being successful (choosing how to interfere) and sets about enacting their own plan.

2.2.1 Simulating Intent Recognition. Although within the narrative generator, the protagonists goal is known, we hide this information from the antagonist. Instead the antagonist is required to observe the protagonist's actions, along with the viewer [1], and then must deduce the most likely goal of the protagonist. Of course they may only be able to observe a certain subset of the protagonist's full action sequence and it is this partial sequence that the antagonist uses to predict the goal of the protagonist. In particular, we would like the generated narratives to reflect the necessary reasoning that the antagonist must do, so that the antagonist can only act to interfere if it is reasonable that they can have guessed the protagonist's intentions. In order to achieve this we have set the problem as a goal recognition task, where the returned likelihood scores of the goal recognition system are used in order to determine the antagonist's certainty in the protagonist's intentions. From the viewer's point of view this leads to appropriate behaviour on the part of the antagonist, as the antagonist is seen to be acting on information that has been presented to them (i.e. the viewer knows what the antagonist knows).

2.2.2 Choosing How to Interfere. Once the antagonist has determined the goal of the protagonist, they will want to interfere. In order to do this they will decide how best to interfere in the protagonist's plan and then they will make a plan to interfere.

The starting point for choosing how to interfere in ProVant is to identify those things that must occur on any plan to the protagonist's goal. Once these are identified we know (and therefore the antagonist might reason) that if the protagonist is aiming to achieve the assumed goal then their plan must achieve these facts. These therefore provide appropriate suggestions for points at which to interfere. After Pozanco et al [21] we view these as fact landmarks [8] (facts that must become true on any path to the goal) and are thus able to exploit efficient algorithms for their automated extraction in ProVant.

We make a distinction between types of antagonist interference: we say they are **recoverable** if the protagonist is still able to achieve their original goal; else they are **non-recoverable** (we define these terms in section 3.1). In particular, it is important that as far as possible that narratives are constructed using recoverable interferences during its main body. Only as a final step and in special circumstances of narrative generation (i.e., if the protagonist's eventual failure is the desired ending) should a non-recoverable interference be considered.

2.3 Narrative Generation with ProVant

The approach implemented in ProVant is to generate narratives displaying the classical hollywood structure by chaining together the following phases: an initial phase of narrative *Exposition*; followed by a series of Protagonist versus Antagonist *Episodes*; and a final phase of goal *Resolution*.

The *Exposition* gives an introduction to the Protagonist and their goal and introduction to the Antagonist and their motivation. As we use a planning approach this information comes directly from the initial state and goal condition of a problem instance (e.g. as shown in Figure 1). The series of *Episodes* that form the backbone of the narrative are formed from segments of protagonist progress towards their goal alongside attempted antagonist interference. If all antagonist interferences are recoverable then the final episode of the narrative is the *Resolution*: a plan for the protagonist that results in them achieving their original goal (this is the case for the narrative shown in Figure 1).

3 THE PROVANT SYSTEM

3.1 Planning Background and Definitions

A PDDL [13] planning problem is a tuple, $\langle \mathbf{P}, A, s^{INIT}, g \rangle$, with a set of propositions, $\mathbf{P}$, a set of actions, A, an initial state, s^{INIT} and set of goal propositions, g. For simplicity we assume that actions are represented by three sets of propositions: the precondition and the add and delete effects. A solution to the planning problem is a plan, π, which is a sequence of actions that transition from s^{INIT} to some state, s_g, that satisfies the goal i.e. $g \subseteq s_g$. Two actions, a_0 and a_1 can be applied simultaneously, denoted, $\langle a_0; a_1 \rangle$, when they do not have interfering conditions and effects (i.e., mutex [2, 21]).

For this work we use a modified problem description: $\langle \mathbf{P}, A^P, A^A, s^{INIT}, \mathbb{G}^{\mathbb{P}}, g^P \rangle$, which separates the protagonist and antagonist's action sets, A^P and A^A, respectively. We notice that the actions in A^P have an additional property that indicates whether the action is observable or hidden and we use this to define a function,

[1] We use viewer by default to refer to an audience of some form of visual media but this could equally be a reader of a text based narrative.

$public(a)$, which holds for observable actions. This function is used when passing actions to the goal recognition system. $\mathbb{G}^{\mathbb{P}}$ defines the set of possible protagonist goals and g^P is the actual goal for this problem. Antagonist goals are identified automatically by the system.

The goal recognition problem takes the input: $\langle \mathbf{P}, A, s^{INIT}, \mathbb{G}^{\mathbb{P}}, obs\rangle$ and returns a likelihood estimate for each $g \in \mathbb{G}^{\mathbb{P}}$, given the set of observations, obs. A threshold value, ψ, is used to indicate when one goal is deemed sufficiently more likely.

As discussed earlier, in order to interfere the antagonist must first determine the protagonist's intended goal:

Definition 1. *The antagonist's current belief about the protagonist's intended goal, $\mathbb{B}_{ANT}(g^P)$, is defined as $g \in \mathbb{G}^{\mathbb{P}}$, if the likelihood of g exceeds the other goals by the threshold, ψ, given the starting state and current set of observable protagonist actions: $\{a | public(a) \wedge a \in a_0^P, \ldots, a_n^P\}$. It is not defined otherwise.*

Once the antagonist has determined the protagonist's goal (believes they have) they then work to make a plan to interfere.

Definition 2. *An interference is a plan, π_A, which, interferes with the protagonist's plan, π^P.*

As noted in [21] the landmarks of a goal are those facts that must become true on any plan that achieves the goal. We therefore know that if the protagonist is aiming to achieve the believed goal then their plan must achieve these facts (these are fact landmarks as introduced by Hoffmann et al [8]). These therefore provide appropriate suggestions for points at which to interfere. For a given landmark, l, for the current believed goal, $\mathbb{B}_{ANT}(g^P)$, the following set of interferences are considered by ProVant:

Definition 3. *The set of candidate interferences, I, is the set of all plans, π^A, that will remove a proposition, p, that is a precondition of an l (a landmark) achieving action.*

If the antagonist has recognised the goal correctly (i.e. $\mathbb{B}_{ANT}(g^P) = g^P$) and chosen an appropriate way to interfere in the protagonist's plan then they will successfully prevent the protagonist from achieving their goal. At this point it is necessary to make a distinction between recoverable and non-recoverable interferences.

Definition 4. *An interference is a* **recoverable interference** *(RI) if the protagonist can recover from the interference and still achieve their goal.*

For an interference $ip \in I$, given the antagonist's plan, $a_0^A, \ldots, a_{n-1}^A$ to interfere with (achieve) ip and the protagonist's current plan, $a_0^P, \ldots, a_m^P$; the projected start state for the protagonist after the interference is s_n=$\texttt{apply}(s_0, a_0^A; a_0^P; \ldots; a_{n-1}^A)$; then ip is recoverable if: $\langle s_n, g^P\rangle$, is solvable by the protagonist (using actions in $\mathbb{A}^P$).

In particular, a narrative should be constructed using recoverable interferences during its main body. Only as a final step and in special circumstances of narrative generation (i.e., if the protagonist's eventual failure is the desired ending) should a non-recoverable interference be considered.

3.2 Algorithm/Pseudo code

The pseudo code for the ProVant system are presented in Figure 2. At the heart of our approach is the generation of interference

```
function PROVANT(s^INIT, g^P)
    s ← s^INIT; π^{P∪A} ← ϵ
    while s ⊭ g^P do
        // extend narrative with choice of episode option
        π_0, ..., π_n ← createEpisodes(s)
        π^ep ← selectEpisode(π_0, ..., π_n)
        s ← s(π^ep); π^{P∪A} ← π^{P∪A} + π^ep
    end while
    return π^{P∪A}
end function

function createEpisodes(s^INIT, g^P)
    i ← 0; Π^eps ← []
    π^P ← getPROPlan(s, g^P)
    // Pro plan plays out while Ant observes
    for all i ∈ 0, ... |π^P| do
        // When sure, Ant's possible interferences are explored
        π^pre ← π^P_0, ..., π^P_{i-1}
        if (ANTsureOfMotive(s^INIT, π^pre)) then
            s ← simulateSteps(s^INIT, π^pre)
            for all g^A ← ANTInterferences(s^INIT, π^pre) do
                π^ep ← enactInterference(s, π^P, i, g^P, g^A))
                Π^eps.add(π^pre + π^ep)
            end for
        end if
    end for
    return Π^eps
end function

function enactInterference(s, π^P, i, g^P, g^A)
    j ← 0; π^ep ← ϵ
    π^A ← getANTPlan(s, g^A)
    // Pro and Ant plans are played out simultaneously
    while (s ⊭ g^P ⋀ s ⊭ g^A ⋀ applicable(s, ⟨π^A_j; π^P_i⟩)) do
        s ← simulateStep(s, ⟨π^A_j; π^P_i⟩)
        π^ep ← π^ep + ⟨π^A_j; π^P_i⟩; i ← i + 1; j ← j + 1
    end while
    // Continue applying Ant plan if applicable
    while (s ⊭ g^A ⋀ applicable(s, π^A_j)) do
        s ← simulateStep(s, π^A_j)
        π^ep ← π^ep + π^A_j; j ← j + 1
    end while
    return π^ep
end function
```

Figure 2: Pseudo code for ProVant. The ProVant function iteratively grows a narrative by generating a set of interference episodes and then selecting one to continue the narrative. Each episode (`createEpisodes`) starts from the protagonist generating a plan to their goal. The antagonist has the opportunity to observe the protagonist's (public actions) progress and attempts to guess their intentions. Where their intentions can be guessed, the set of possible interference is generated and each is used to construct an interference episode (`enactInterference`).

episodes, where the protagonist attempts to achieve their goal and the antagonist identifies a way of interfering with the protagonist so that they are prevented or delayed. A narrative is constructed of one or more episodes until the protagonist finally achieves their goal. At the top level the **PROVANT** function is responsible for growing the narrative by selecting a sequence of interference episodes, which will lead to the protagonist eventually achieving their goal. The first stage is to generate the set of interference episodes for the current state, as described below. The second step is to select one to continue the narrative. Once one has been selected the state and current narrative fragment is updated and the process is repeated.

The selection between the proposed episodes provides an opportunity to control how the narrative is progressed and there may be several criteria that are used at this point in order to make this decision. In this work we prefer those episodes (if any) where the antagonist has blocked the protagonist and where the protagonist can still recover. In order to explore several interference episodes we then select the episode that leaves the protagonist the furthest from their goal (measured as the length of a plan from the final state of the episode to their goal).

Each episode starts from the protagonist generating a plan in an attempt to achieve their aims (`createEpisodes`). At this point we address the problem that the antagonist has in terms of determining the intentions of the protagonist (indicated by the function `AntsureOfMotive`). This method attempts to determine the intentions of the protagonist by utilising a goal recognition system. For a given initial state, set of observations, and a set of potential goals, the system attributes a heuristic likelihood score to each of the potential goals. Our system then uses these scores to determine whether the antagonist can guess the protagonist's goal. We have used the approach presented in [17] as it can operate from partial observation sequences. This allows us to more accurately reflect the reasoning process by only providing the goal recognition system with the actions that the antagonist can have observed.

Each of the points in the plan where the protagonist's intentions can be guessed provides an opportunity to explore the possible interferences of the antagonist. For example, the antagonist may begin acting as soon as they can, or there might be some delay, which might change how they will end up interfering. The `createEpisodes` method gathers all of the possible interferences for each of these points. The set of possible interferences is generated using a process similar to [21]. The first step is to select the antagonist's belief of the protagonist's goal (the one that the goal recognition indicated was most likely) and then generate the set of landmarks for this goal. The system then identifies the set of interferences. We assume in our pseudo code that each of the generated interferences is actable on by the antagonist [2].

Once a starting point and an interference have been selected the `enactInterference` method generates the interference episode by attempting to execute the plans together. Where there is a conflict between the actions, the antagonist's action is enacted. Once the protagonist or antagonist have achieved their goal, or indeed one of them is blocked then any remaining actions of the antagonist's plan are applied and the episode is returned.

[2]In practice, this property can be hard to calculate up front and we instead prune during the enaction of the interference.

4 WORKED EXAMPLE

To illustrate the process of narrative generation with ProVant consider generation of a narrative for the Senior School domain introduced in section 2; the key steps of the process are illustrated in Figure 1. We will assume that the ProVant system is called with the goal that John, the protagonist, wants to go to prom with Ann. And that Tom, the antagonist, believes that John either wants to join the sports team, or otherwise take Ann to the prom. The system starts with a shared state, which captures the initial positions and attributes of the characters (e.g., John has an instrument). As John's goal is not yet achieved a collection of plausible episodes is generated (`createEpisodes`). The first step is for a plan to be constructed that allows John to achieve his goal. An example plan is presented in actions 1-8 of Figure 1. Note that narratives must begin with initial steps of exposition of the goal of the protagonist (action 1) and the motivation of the antagonist (action 2). As discussed earlier (section 2.3).

Our system then steps through the protagonist's plan; starting from (introduce-pro john). At each step the set of observable actions (from the beginning of the episode) are identified and used to determine whether the antagonist can be certain of the protagonist's aims. In this case the first two actions are not observable, providing no clues to John's intentions. However, if we assume that at the third step Tom can observe John joining a band then the goal recognition system returns the following values for the two goals:

- ➔ (and (go-prom john ann)): 0.2857143
- ➔ (and (on-team john)): 0.16666667

The goal recognition approach returns a heuristic value that indicates the likelihood of the goal being the intended goal. They use a threshold value to determine when the difference in heuristic estimates is sufficient in order to select a single goal. A difference of 0.2 between the values returned by the goal recognition is deemed sufficient to indicate that the protagonist's goal is clearly distinguishable. As such, the antagonist is still not certain at this step.

However, the observation of John playing in the band changes the estimates provided by the goal recognition system:

- ➔ (and (go-prom john ann)): 0.42857143
- ➔ (and (on-team john)): 0.16666667

In this case, the antagonist is certain of the protagonist's goal (method `AntsureOfMotive` returns True) and can consider interfering. The set of landmarks is generated for John's believed goal:

(go-prom john ann): [..., (attractive john), (crush ann john),
(working-car john),
(asked-to-prom john ann),...]

The system then generates the set of actions that the protagonist can use to achieve each of these propositions. For example, the action (go-see-band Ann), has the effect of (crush ann john). The preconditions for these actions are then examined in order to determine if the antagonist can delete them. At this stage the system identifies that the antagonist can delete the (has-instrument john) precondition of (go-see-band Ann), by applying the (steal-instrument tom john) action. It also identifies that the (car-working john) precondition of the (drive-to-prom john ann) action can be deleted with the (break-car tom) action.

Each of these interferences are explored, by calling the `enactInterference` method, leading to the construction of a possible episode. Figure 1 presents two instances of this method: action 9 and action 16 (the latter is from a later part of the narrative construction process). In these cases the antagonist blocks the protagonist from taking an action that is important for their goal. For example, in stealing John's instrument, Tom prevents John from being seen in the band.

After all of the feasible interferences have been examined at this step, the system continues stepping through the protagonist's plan, considering each step as a possible starting point for the antagonist's interference. In the proceeding steps the antagonist's certainty in the protagonist's intentions increases, e.g., after John has asked Ann to the prom (action 6):

➔ (and (go-prom john ann)): 0.71428573
➔ (and (on-team john)): 0.16666667

At this point the set of landmarks that have not been achieved is smaller, resulting in fewer opportunities to interfere. The set of open landmarks is:

(go-prom john ann): [...,(working-car john),
(asked-to-prom john ann),...]

Tom can now only interfere with the car-working precondition of the drive-to-prom action.

The complete set of single interference episodes is returned to the ProVant method. In this example, all of the episodes are recoverable and therefore none are pruned. For example, Figure 1 presents one of the possible continuations of the protagonist's plan after their instrument is stolen. The system selects the narrative that leaves the protagonist furthest from their goal. In this example, this is achieved by selecting the episode where Tom steals John's instrument (forcing John into a longer sequence of actions in order to play with the band). The effect of the episodes are enacted on the state and the process is repeated. In this case the protagonist will need to discover some way of compensating for the antagonist's interference. As we observed above, as part of the narrative has already taken part there are fewer open landmarks and therefore fewer opportunities to interfere. Moreover, our system ensures that if the protagonist can achieve their goals then a narrative is constructed. It achieves this by pruning non-recoverable episodes and not allowing the antagonist to interfere in the same way twice.

5 EVALUATION

For the evaluation we created a test set of PDDL models for the following narrative domains, all of which are based on examples from the literature:

- Aladdin [24]. The familiar folk tale set in a magical land.
- Little Red Riding Hood [23]. Based on the European fairy tale.
- Western [33]. Featuring a rancher called Hank and a railroad developer who wants his land.
- Raiders of the Lost Ark [33]. Featuring an archaeologist who wants to find a lost treasure and others who act to prevent him.
- Prom Week [12]. A high school story domain with characters wanting to gain social status, get a date for the prom and so on.
- Detective domain [10]. A detective wants to solve crimes and get promoted. A thief wants to commit crime and escape detection.

Domain	$\lvert\pi^P\rvert$ (no ant)	$\lvert\pi^P_{\mathbb{I}}\rvert$ (with ant)	$\lvert\pi^{A\cup P}_{\mathbb{I}}\rvert$	#INT
Aladdin	7	11	16	3
Basketball	12	14	16	2
Raiders	8	9	11	3
RedHood	3	5	7	2
School	8	11	13	2
Western	8	11	13	2

Table 1: This table records several properties for the narrative selected by ProVant on a collection of story domain problems. It records the length of the protagonist's plan with no interference ($\lvert\pi^P\rvert$) and with interference ($\lvert\pi^P_{\mathbb{I}}\rvert$). It records the overall narrative length, including both agents ($\lvert\pi^{A\cup P}_{\mathbb{I}}\rvert$) and the number of times the antagonist interferes (#INT).

The domain actions were partitioned depending on their relevance for the different types of agent (protagonist or antagonist) with some actions of relevance to both (as well as other agents in the world). This is similar to strategies used in other narrative approaches: ImpractIcal used an `actor` flag in its PDDL action specification to constrain the action to character agents of the desire type [31]; and similarly glaive used a PDDL representation augmented with an `agents` field, denoting agent relevance [33].

Each domain supports a variety of goals, which means that the goal recognition task is not trivial. For example, in the School domain, the protagonist can have different goals such as wanting to make the school team, or take another student to the prom.

5.1 Experiments

The system was implemented in Python and exploits metric-ff [7] and an implementation of the goal recognition approach presented in [17]. As metric-ff is not an agent based planner, when planning for a particular agent, the relevant action set for that agent is constructed and provided to the planner.

We wanted to demonstrate that ProVant is capable of automatically "complexifying" narrative (as termed by Riedl [23]) without the need for authored goals. To this end the system was used to generate narratives for each of the 6 domains described above. Table 1 indicates the impact of the antagonist interferences on the overall length of the narrative. Column $\lvert\pi^P\rvert$ details the length of a narrative generated to achieve the protagonist's original goal. In this case there are no interference from the antagonist. The next columns present the number of narrative actions in the complete narrative, which includes antagonist interferences. $\lvert\pi^P_{\mathbb{I}}\rvert$ presents the number of protagonist actions and $\lvert\pi^{A\cup P}_{\mathbb{I}}\rvert$ presents the total narrative length for both agents (i.e., the length of the overall narrative chain). The results indicate that the antagonist's actions have an important impact on the protagonists path to achieving their goal. In particular, the protagonist must extend or change their plan to compensate for the antagonist's interferences. In these cases the average extension of the narratives from the base-line (no antagonist interferences) is 62%. The nature of how the antagonist's actions alter the generated narrative vary between the domains. For example, in the School domain, the antagonist might steal the protagonist's instrument, preventing, or delaying the protagonist

Domain	$\lvert eps \rvert$	$\lvert eps^A \rvert$	$\lvert eps^{NR} \rvert$	$\lvert eps^{\setminus 0} \rvert$
Aladdin^{All}	38	30	0	8
Aladdin^{Pub}	37	30	0	7
Basketball^{All}	24	18	0	3
Basketball^{Pub}	21	9	0	5
Raiders^{All}	13	5	3	7
Raiders^{Pub}	10	3	1	6
RedHood^{All}	9	4	0	5
RedHood^{Pub}	7	3	7	4
School^{All}	12	9	0	3
School^{Pub}	12	9	0	3
Western^{All}	9	6	0	3
Western^{Pub}	10	5	0	4

Table 2: Table records number of episodes generated while constructing the narrative for each of the domains, with full observability (domain^{All}) and where antagonist can only observe public actions (domain^{Pub}). The table records the number of episodes that are won by the antagonist ($\lvert eps^A \rvert$), that cannot be recovered from ($\lvert eps^{NR} \rvert$) and that finish the narrative ($\lvert eps^{\setminus 0} \rvert$): those where the protagonist reaches their goal.

from joining a band. The protagonist must then reconsider how to attract Ann's attention.

The final column indicates the number of antagonist interferences during the narrative. This demonstrates that the ProVant system is able to chain together several protagonist/antagonist episodes.

Table 2 presents the number and categories of episodes that are generated during the construction of the narratives. The number of episodes generated provides an indication of the diversity of narrative structures that are supported by the system. In particular, for each episode selected during narrative construction, there were an average of 5 alternatives to choose from.

The table shows that in the Raiders of the Lost Ark (Raiders) and Little Red Riding Hood (RedHood) the system generated non-recoverable interferences. Because these are identified these provide an opportunity for controlling the narrative structure, without accidentally discovering a dead-end.

5.2 Authoring Reduction

To assess the reduction in authoring effort we use the number of antagonist interferences in ProVant narratives as a proxy for authored structuring goals. Column #INT in Table 1 shows the number of times the antagonist interfered for our test set of problem instances across the selected narrative domains: on average 2-3 interference points for plans containing 7-16 actions. This appears roughly comparable to the number of authored narrative structuring goals reported in the literature. For example, [23] includes examples with 2 author goals per plans of 8-10 actions in length, while [20] includes figures which suggest a minimum of 1-2 authored goals to every 10 action per output narrative plan for average length plans of 65-75 actions. This is a significant reduction in authoring effort as ProVant removes the need to specify these.

(1) John is a geek. He has a crush on Ann. John announces he wants to take Ann to the Prom. (2) Tom hates John. He wants to stop John being successful. (3) John joins a band so he doesn't look like a geek. (4) John plays with the band. (9) Tom steals John's instrument so he can't be seen playing with the band. (10) John gets a job washing dishes to earn money to buy a new instrument. (11) John buys an instrument with the money. (12) Ann goes to see the band and develops a crush on John. (13) John asks Ann to the prom. (16) Tom sneaks in and breaks John's car so he can't drive Ann to the prom. (17) John borrows a car as his is broken. (18) John drives Ann. (19) John and Ann go to the prom.

Figure 3: Example Senior School Narrative: the ProVant narrative plan in Figure 1 has been translated to text sentences (numbers in brackets correspond to actions in the plan).

(1) John is a geek. He has a crush on Ann. John announces that he wants to take Ann to the Prom. Sue is a cheerleader. Sue doesn't want a date to the Prom. (3) John joins a band so he doesn't look like a geek. (4) John plays with the band. Sue goes cheerleading. (12) Ann goes to see the band and develops a crush on John. Sue goes shopping for a Prom dress. (13) John asks Ann to the Prom. (18) John drives Ann. (19) John and Ann go to the Prom.

Figure 4: Example School Narrative for Validation: non-interfering actions (underlined) replace antagonist interference (numbers in brackets remain from plan in Fig. 1).

We observe that it is likely that different sets of authored goals would be required for different types of narrative goals so the number required would be much larger than this in practice. For example, [19] had 4 different goal themes and we could reasonably assume that different types of goals would require different collections of goal structuring information. Thus the numbers of authored goals required would quickly escalate. In contrast ProVant automatically identifies these interference points thus removing entirely the overhead of authoring them as an additional part of the model.

5.3 User Study

We also conducted a study to evaluate if users were able to recognise the structure of ProVant narratives: both the role of the Protagonist and Antagonist agent; and also the interferences enacted by the Antagonist. For the study we recruited 39 English speaking adults who completed an on-line questionnaire. We used the same domains as the experiments (Aladdin, Basketball, Raiders, Red Riding Hood, Senior School, Western), however to mitigate for prior knowledge of Aladdin, Raiders and Red Riding Hood names of characters and objects were mapped to alternates as follows:

Aladdin: Aladdin → Jack, Jasmine → Fiona, Jafar → Hook
Raiders: Indiana → Prof. Brown, Nazi → Col. Grey, Ark → Treasure
RedHood: Red Riding → George, Bad Wolf → Snake, Granny → Uncle

Narrative plans were translated to text sentences for use in the study. As illustration, Figure 3 shows the text translation of the ProVant plan used in the worked example (section 4). At the start of the survey users were told the role of a Protagonist ("the main character who tries to achieve some goal") and Antagonist ("another character, who tries to obstruct them"). Then for each story they were asked to name the Protagonist and the Antagonist if they thought any of the characters filled this role. They were also asked if they thought the Antagonist had interfered and if they responded "yes" then they were asked to state all of the ways in which they had interfered.

Domain	Protagonist	Antagonist	#Interferences
Aladdin	100	100	86.5 (13.5)
Basketball	97.4	97.4	83.8 (16.2)
Raiders	100	97.4	65.8 (34.2)
Red Riding Hood	97.4	97.4 %	80.6 (19.4)
Senior School	94.9	100	97.2 (2.8)
Western	100	100	94.6 (5.4)

Table 3: Results of User Study (shown as %). Users were shown text translation of narrative plans and asked to identify Protagonist, Antagonist and #Antagonist interferences. The Interferences column shows the % who correctly identified all interferences with partially correct in brackets.

The results of the survey are shown in Table 3. It can be seen that users were clearly able to identify the Protagonist and Antagonist. Further they were able to identify the antagonist interferences.

As comparison to validate the approach we also showed users a narrative without a clear Antagonist for the Senior School domain which is the narrative shown in Figure 4. This narrative was created by replacement of interfering actions with other actions for an unrelated character. Thus the variants used in this study were N1 (Figure 3 and N2 (Figure 4). For both variants users were asked to identify the Protagonist and Antagonist if they think there is one. For N2, all users correctly reported that there was no Antagonist. Users were also asked if they found either of the narratives more interesting: all users selecting N1 as more interesting and offering free text explanations that it was the antagonist interference as explanation. The following give a flavour of the free-text comments offered by users: *"In story #1, the antognist provided a more interesting story than story #2", " More happened to obstruct John", "More drama due to the antagonist", "There are conflicts".*

6 RELATED WORK

A popular approach in narrative generation has been to use some form of "author goals" as a mechanism to help structure output narratives. These author goals can be seen as forming a scaffold on which the narrative can be built by specifying intermediate states for the narrative to pass through and to help shape its trajectory. This helps to ensure that the story meets desired quality criteria (i.e. the view of the author/director/designer) and also to help add complexity to the narrative. Plan-based approaches which have used this strategy include the approach of Riedl [23] who used author goals to encode desired narrative properties and "complexify" the output narrative: partial-order planning being used for generating the story and the author's goals incorporated as constraints on the generated results. Porteous et al [18] used a similar mechanism with their narrative "constraints" which were partially specified state descriptions, used to shape narrative trajectories.

With non plan-based approaches similar structuring mechanisms have also been used. For example: both Lamstein et al [11] and Nelson et al [15] used author events to help shape narratives in their interactive systems; whilst the THESPIAN system [28, 29] used directorial goals to control structure in the interactive drama.

Regardless of the algorithmic approach, these narrative generation approaches require the specification of author goals (along with any other orders between goals) which places additional burden on the task of authoring these generative narrative models. In contrast ProVant removes this part of the authoring task as these intermediate states which help shape and complexify the narrative are automatically identified by the system.

There has been some calls for the use of adversarial search techniques to be extended beyond game AI to narrative and storytelling. For example, Roberts el al [25] propose an adversarial model in which agents encounter problems in achieving their goals, which obstruct them and forces them to behave in ways other than those which the agent originally intended. This is similar to the classical hollywood structure [3] which we have targeted with ProVant. Whilst there has been some exploration of these ideas in game like settings to enhance player experience there is still a requirement for the inclusion of author goals. For example, Roberts et al [26] evaluated mechanisms for automated story shaping and endeavoured to reduce the authoring required to shape players' interactive narrative experiences. They explored the use of more abstract author goals which, whilst leveraging some reductions in the scripting burden, nevertheless did not remove the need for them completely.

The ProVant system builds on recent results of Pozanco et al [21] who presented an automated approach to counter planning using goal recognition and landmarks. We use a planning-based goal recognition approach similar to [22] which assumes observations are actions and hence removes the need for plan libraries and automated identification of fact landmarks as part of antagonist interference reasoning [8]. Whilst drawing inspiration from Pozanco et al [21], ProVant differs in a number of important respects. we introduce a different mechanism for selection of interference points (over their single Counterplanning goals); Importantly, ProVant has a bias away from non-recoverable interference whereas with counterplanning, the generated counterplanning goals are vital to stop the seeking agent from achieving their goal (non-recoverable). Further, with ProVant the protagonist's goal is known, so goal recognition is only used as a proxy for antagonist reasoning. This ensures that interference occurs at points that appear reasonable to the viewer i.e., could reasonably be deduced by the antagonist.

7 CONCLUSION

We have presented a novel mechanism for generating narratives displaying the "classical hollywood" structure. The approach is fully automated and based on: goal detection (antagonist recognition of protagonist intent); automated identification of interference points (landmark extraction); and plan generation (protagonist plan to goal; antagonist interference). Results show our approach can automatically identify interference points and use those to complexify the narrative in keeping with the target structure. Importantly, automated interference identification removes the need for authored narrative structuring information. Results of a study showed users can recognise narrative structure and agent roles: thus evidencing that system narratives display the target structure.

We have viewed the protagonist and antagonist as system controlled virtual agents but future work could extend the approach to more game like settings where the viewer can take on the role of protagonist or antagonist. As part of this the generation mechanism could be used to assist in the creation of the planning domain models: using goal recognition and landmark extraction as the model is created to identify potential sources of antagonist interference.

REFERENCES

[1] R. Aylett, J. Dias, and A. Paiva. An Affectively Driven Planner for Synthetic Characters. In *Proceedings of 16th Int. Conference on Automated Planning and Scheduling (ICAPS)*, 2006.

[2] A. Blum and M. Furst. Fast Planning through Planning Graph Analysis. In *Proceedings of IJCAI*, 1995.

[3] D. Bordwell. *Narration in the Fiction Film.* Madison: University of Wisconsin, 1985.

[4] J. G. Carbonell. Counterplanning: A strategy-based model of adversary planning in real-world situations. *Artif. Intell.*, 16(3):295–329, 1981.

[5] A. J. Champandard. The Evolution Of Planning Applications and Algorithms in AAA Games, 2014. AI Game Dev (available online: http://aigamedev.com/premium/interview/planning-analysis/ [accessed: 31-10-2018]).

[6] Y.-G. Cheong and R. M. Young. Suspenser: A Story Generation System for Suspense. *IEEE Transactions on Computational Intelligence and Artificial Intelligence in Games*, 7(1), 2015.

[7] J. Hoffmann. The Metric-FF Planning System: Translating "Ignoring Delete Lists" to Numeric State Variables. *Journal of Artificial Intelligence Research*, 20:291–341, 2003. Software download: http://fai.cs.uni-saarland.de/hoffmann/metric-ff.html.

[8] J. Hoffmann, J. Porteous, and L. Sebastia. Ordered Landmarks in Planning. *Journal of Artificial Intelligence Research (JAIR)*, 22:215–278, 2004.

[9] S. Kambhampati. Model-lite Planning for the Web Age Masses: The Challenges of Planning with Incomplete and Evolving Domain Models. In *Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI*, 2007.

[10] B. Kartal, J. Koenig, and S. J. Guy. User-Driven Narrative Variation in Large Story Domains using Monte Carlo Tree Search. In *Proceedings of the 13th Int. Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, 2014.

[11] A. Lamstein and M. Mateas. A Search-based Drama Manager. In *AAAI Workshop Series: Challenges in Game Artificial Intelligence*, 2004.

[12] J. McCoy, M. Treanor, B. Samuel, A. A. Reed, M. Mateas, and N. Wardrip-Fruin. Social Story Worlds With Comme il Faut. *IEEE Transactions on Computational Intelligence and AI in Games*, 6(2), 2014.

[13] D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins. PDDL - the Planning Domain Definition Language. Technical report, CVC TR-98-003/DCS TR-1165, Yale University, 1998.

[14] R. McKee. *Story: substance, structure, style, and the principles of screenwriting.* NY: ReganBooks, 1997.

[15] M. J. Nelson, D. L. Roberts, C. L. Isbell, and M. Mateas. Reinforcement Learning for Declarative Optimization-based Drama Management. In *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2006.

[16] J. Orkin. Three States and a Plan: The AI of F.E.A.R. In *Proceedings of the Game Developer's Conference (GDC)*, 2006.

[17] R. Pereira, N. Oren, and F. Meneguzzi. Landmark-Based Heuristics for Goal Recognition. In *Proceedings of 31st Conference on Artificial Intelligence (AAAI)*, 2017. Software download: https://github.com/ramonpereira/Planning-GoalRecognition.

[18] J. Porteous, M. Cavazza, and F. Charles. Applying Planning to Interactive Storytelling: Narrative Control using State Constraints. *ACM Transactions on Intelligent Systems and Technology (ACM TIST)*, 1(2):1–21, 2010.

[19] J. Porteous, F. Charles, and M. Cavazza. NetworkING: using Character Relationships for Interactive Narrative Generation. In *Proceedings of 12th Int. Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2013)*, pages 595–602. IFAAMAS, 2013.

[20] J. Porteous, F. Charles, C. Smith, M. Cavazza, J. Mouw, and P. van den Broek. Using Virtual Narratives to Explore Children's Story Understanding. In *Proceedings of 16th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*. IFAAMAS, 2017.

[21] A. Pozanco, Y. E-Martiń, S. Fernańdez, and D. Borrajo. Counterplanning using Goal Recognition and Landmarks. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI-18*, 2018.

[22] M. Ramírez and H. Geffner. Probabilistic Plan Recognition Using Off-the-Shelf Classical Planners. In *Proceedings of the 24thConference on Artificial Intelligence (AAAI)*, 2010.

[23] M. Riedl. Incorporating Authorial Intent into GenerativeNarrative Systems. In *Proceedings of AAAI Spring Symposium on Intelligent Narrative Technologies*, 2009.

[24] M. O. Riedl and R. M. Young. Narrative Planning: Balancing Plot and Character. *Journal of Artificial Intelligence Research*, 39:217–267, 2010.

[25] D. Roberts, M. Riedl, and C. Isbell. Beyond adversarial: The case for game AI as storytelling. In *Proceedings of the Conference of the Digital Games Research Association*, 2009.

[26] D. L. Roberts and C. L. Isbell. Lessons on Using Computationally Generated Influence for Shaping Narrative Experiences. *IEEE Transactions on Computational Intelligence and AI in Games*, 6(2), 2014.

[27] D. E. Rumelhart. On Evaluating Story Grammars. *Cognitive Science*, 4:313–316, 1980.

[28] M. Si, S. Marsella, and D. Pynadath. Evaluating Directorial Control in a Character-Centric Interactive Narrative Framework. In *Proceedings of the 9th Int. Conference on Autonomous Agents and Multiagent systems (AAMAS-10)*. IFAAMAS, 2010.

[29] M. Si, S. Marsella, and D. V. Pynadath. Thespian: using multi-agent fitting to craft interactive drama. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M. P. Singh, and M. Wooldridge, editors, *Proceedings of 4th Int. Conference on Autonomous Agents and Multiagent Systems (AAMAS 2005)*, pages 21–28, Utrecht, The Netherlands, July 2005.

[30] S. Spielberg. Lucasfilm, 1981. Raiders of the Lost Ark.

[31] J. Teutenberg and J. Porteous. Efficient Intent-based Narrative Generation Using Multiple Planning Agents. In *Proceedings of 12th Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, 2013.

[32] P. W. Thorndyke. Cognitive Structures in Comprehension and Memory of Narrative Discourse. *Cognitive Psychology*, 9:77–110, 1977.

[33] S. G. Ware and R. M. Young. Glaive: A State-Space Narrative Planner Supporting Intentionality and Conflict. In *Proceedings of the 10th Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*, 2014.

[34] R. M. Young. Notes on the use of plan structures in the creation of interactive plot. In *AAAI Fall Symposium on Narrative Intelligence*, 1999.

[35] R. M. Young. Creating Interactive Narrative Structures: The Potential for AI Approaches. In *AAAI Spring Symposium in Artificial Intelligence and Entertainment*. AAAI Press, 2000.