The AIIDE-17 Workshop on
Intelligent Narrative Technologies
WS-17-20

Planning Graphs for Efficient Generation of Desirable Narrative Trajectories

Adam Amos-Binks
Liquid Narrative Group and
Department of Computer Science
NC State University
Raleigh, NC, USA, 27695
aaamosbi@ncsu.edu

Colin Potts
Liquid Narrative Group and
Department of Computer Science
NC State University
Raleigh, NC, USA, 27695
cmpotts@ncsu.edu

R. Michael Young
Liquid Narrative Group and
School of Computing and
The Entertainment Arts and Engineering Program
University of Utah, Salt Lake City, UT, USA 84112
young@eae.utah.edu

Abstract

A goal of Experience Managers (EM) is to guide users through a space of narrative trajectories, or story branches, in an Interactive Narrative (IN). When a user performs an action that deviates from the intended trajectory, the EM uses a mediation strategy called *accommodation* to transition the user to a new desirable trajectory. However, generating the trajectory options then selecting the appropriate one is computationally expensive and at odds with the low-latency needs of an IN. We define three desirable properties (exemplar trajectories, narrative-theoretic comparison, and efficiency) that general solutions would possess and demonstrate how our plan-based Intention Dependency Graph addresses them.

Introduction

Branching story games (e.g. (Telltale Games 2012)) have gained popularity for presenting a narrative that appears to adapt to user actions within a story world. This trend has led to research in branching stories ranging from the creative affordances of beats (Mateas and Stern 2005) to more formal story-plans in automated planning (Porteous, Cavazza, and Charles 2010). All representations share the need for an Experience Manager (EM) to effectively guide a user through a space of desirable narrative trajectories (Riedl and Bulitko 2012), or story branches, in an Interactive Narrative. This can take the form of avoiding certain story events while ensuring others are part of a narrative experience.

When an IN (Interactive Narrative) offers numerous story branches, it can lead to user actions that do not conform to an EM's desired trajectory. An EM can respond to this in two different ways. The first is an *intervention* strategy that simply prevents a user from executing an action that would lead to an undesirable story trajectory. The desirable narrative trajectory –a story-plan in an automated planning context– in Figure 1 is story-plan A, drawn from to the classic Little Red Riding Hood (LRRH). Should a user attempt to move Red Riding Hood (*rr*) to the Bad Wolf's (*bw*) cave

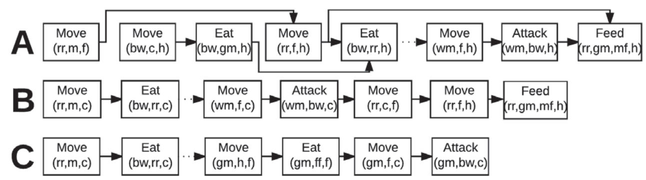

Figure 1: Little Red Riding Hood plan-based trajectories

(*c*), deviating from the desired trajectory where she goes to the forest(*f*), the EM could *intervene* by having the path to *c* blocked, forcing the user to *f*. The second strategy, *accommodation*, is a more involved strategy that allows the user to execute the problematic action, prompting the EM to generate new trajectories that incorporate the new action in a new trajectory. In our example, the EM would incorporate *rr*'s move to *c* by generating trajectories B and C and then choosing one as the new desired trajectory.

While there has been research addressing the intervention strategy of plan-based INs (e.g. (Magerko 2007)), the accommodation strategy has received little attention outside of Ramirez and Bulitko's recent work (Ramirez Sanabria and Bulitko 2014). There are three properties that make supporting accommodation a difficult problem. The first is that the generation process must produce *exemplar* trajectories characterizing the narrative content of story branches so that guarantees can be made about choosing a new trajectory (e.g. fewest character goal changes). Second, generated trajectories must be *comparable* on narrative-theoretic properties so that a new desirable trajectory can be assessed in relation to the previous one (e.g. number of character goal changes). A final property is that trajectory generation must be efficient enough to respond in real-time. Solutions addressing these properties would enable an EM to perform *accommodation* using an optimal narrative trajectory rather than merely a satisfying one (in the boolean sense).

The primary contribution described in this paper is a plan-based algorithm for finding desirable trajectories that addresses the three properties described above. To accomplish this, we extend the planning and goal graphs used by the Glaive state-space narrative planner (Ware et al. 2014) to construct an Intention Dependency Graph (IDG). The IDG captures the necessary co-occurrence of character goals in a set of exemplar trajectories that estimates the narrative content range of story branches in a computationally efficient manner. We demonstrate some preliminary results using a simplified classic tale, Little Red Riding Hood.

Previous Work

A central research goal of the IN community is to balance authorial goals, such as logical plot progression, with those that enable user agency. To address this gap, Bates (Bates 1992) proposed a *drama manager* (DM) to monitor and intervene in order to manage the dramatic experience quality. Building on this foundation, research to design a DM that closely ties a game environment to artificial intelligence that automatically produces action sequences led to the *Mimesis* system (Young and Riedl 2004). This work led to active area of DM research in games summarized in Roberts and Isbell (Roberts and Isbell 2008). A generalization of a DM is the *experience manager* (EM) (Riedl and Bulitko 2012), which relaxes the dramatic requirement on narratives, allowing them to have training or educational applications. Generally, an EM must anticipate the experiences available to a user given the state of the story world. The EM then subtly mediates with the story world to ensure the user remains on a narrative trajectory that adheres to some experience quality.

One mediation strategy used in plan-based EMs to ensure causal coherence is *accommodation*. Accommodation restructures a narrative plan to account for the resulting state after a user has disrupted causal links on the narrative trajectory. In the General Mediation Engine (GME) (Robertson and Young 2015), plan restructuring queries a *system response oracle* for a new narrative trajectory (story-plan) to solve the planning problem resulting from user actions. However, when using a plan-based representation, the computational requirements of solving a planning problem limit the ability of the *system response oracle* to find the most desirable new narrative trajectory. Simply finding a new solution plan is difficult and the first one found is often used, leaving no process to choose the most appropriate trajectory from a set of narrative trajectories. If the system response is a trajectory with a great deal of change in character behavior (old character goals dropped, new ones being pursued for no apparent reason), then the user may become confused or disinterested by the resulting inconsistent character behavior. If the EM were aware of a trajectory set, then one could be chosen that ensured desired outcomes, such as one that minimized changes to character goals.

We address the *exemplar* narrative trajectories property through a fundamental property of intentional plans (Riedl and Young 2010). Intentional plans require that every action in a solution plan be in service of a character goal (happenings are part of *fate*'s goals). Character goals are rarely self-contained, in that they require the effects of other characters' actions to establish states of the world and motivations. We leverage these dependencies to obtain a set of qualitatively different exemplar trajectories, where each trajectory represents a class of intentional plans containing the same conjunction of character goals. Intentional paths from an exemplar's character goals can then be combined to produce candidates for solving a narrative planning problem.

This approach to finding qualitatively different exemplar trajectories differs from the broader planning community's generation of diverse solutions. Generating the solution diversity of a planning problem requires solving it multiple times and relies on the search process to explore all options in a principled manner when searches are often designed to find only minimal solutions. While existing planning systems have made progress towards exploring the solution diversity(Coman and Muñoz-avila 2011; Nguyen et al. 2012; Bryce 2014), it is still an open area of research and is not performed in a narrative-theoretic manner such as by finding combinations of character goals. Determining a problem's solution diversity without the entire computational burden of planning would enable an EM to make more informed accommodation choices.

To ensure that our approach addresses the narrative-theoretic *comparison* property, we leverage intentional plans and plan distance metrics. An intentional plan ensures that all steps in a plan serve character goals, in addition to the authorial goals in the planning problem, and have been implemented by several narrative planners (Riedl and Young 2010; Ware 2014). Due to its non-normal assumptions about the data, Jaccard distance forms the basis of several domain-independent (e.g. (Srivastava, Nguyen, and Gerevini 2007)) and domain-specific (e.g. (Amos-Binks, Roberts, and Young 2016)) plan distance metrics. We use Jaccard distance as the basis of a domain-specific plan comparison metric that compares character goals between exemplars.

Finally, to address the *efficiency* property, we turn to the broader planning community's work in problem formulation and strategies for efficient planning. Specifically, the *planning graph* (Blum and Furst 1997; Bryce and Kambhampati 2007) is used by many state-of-the-art planners (e.g. (Helmert 2006)) to calculate heuristics to speed up the search process. Similarly, the Glaive (Ware and Young 2014) narrative planner uses the planning graph but with additional narrative-theoretic constraints captured with character *goal-graphs*. The goal-graph identifies the action sequences a character could take in support of an adopted goal. While planning and goal graphs are used to prune the search space, we use them to identify the intentional and motivation dependencies between character goals that constrain the solutions to the narrative planning problem, without having to solve the planning problem exactly.

We have outlined the technical challenges of an EM when performing accommodation and described three desirable properties of a solution that would overcome these challenges. To address the three properties using plan-based approach, we reviewed research in intentional planning, plan comparison, and efficient planning from the broader planning community. This research forms the basis of the IDG as a plan-based accommodation solution.

Init. state
at(rr, m)
at(wm, f)
at(gm, h)
at(bw, c)
safe(rr)
safe(wm)
safe(gm)
safe(bw)
hungry(gm)
has(rr, mf)
at(ff, f)

Character	
rr	redriding hood
gm	grandma
wm	woodsman
bw	bad wolf

Goal state
safe(gm)
safe(rr)
not(hungry(gm))

Place	
m	market
h	house
f	forest
c	cave

Character, Goal	
rr	not(hungry(gm))
gm	not(hungry(gm))
gm	safe(rr)
wm	safe(rr)
bw	not(safe(rr))
bw	not(safe(gm))

Object	
mf	market food
ff	forest food

**Without loss of generality to IDG construction, we add domain-specific conditional preconditions and effects in the Eat action to ensure the the canonical LRRH tale (*bw* dresses as *gm to surprise rr*) can be generated by a story planner.

Move (?char, ?fr, ?to)	
at(?char,?fr)	at(?char, ?to)
safe(?char)	not (at(?char,?fr))

Feed(?char, ?eater, ?foo, ?loc)	
at(?char,?loc)	not(has(?c, ?f))
safe(?char)	not(hungry(?eater))
at(?eater,?loc)	
safe(?eater)	
has(?char,?foo)	

Attack(?char, ?vic, ?loc)	
at(?char,?loc)	not(safe(?vic))
safe(?char)	all Character ?c
at(?vic,?loc)	!= ?vic, safe(?c)
not(hungry (?char))	

**Eat(?char, ?foo, ?loc)	
at(?char,?loc)	not(safe(?foo))
safe(?char)	not(hungry (?char)
at(?foo,?loc)	if ?foo=gm & ?loc=h then surprise(rr)
if ?foo=rr & ?loc=h then surprise(rr)	

Figure 2: Story problem on left and domain actions on right (preconditions and effects underneath on left and right, respectively) for LRRH.

Finding Plan-based Narrative Trajectories

We address the three desired solution properties of accommodation by uncovering the underlying dependencies between character goals from the story planning graph. To represent the motivational and intentional dependencies between character goals we define in a new structure, the Intention Dependency Graph (IDG). Using the IDG, we can reason about the narrative content of trajectories more efficiently than incurring the cost of generating many different solutions to a planning problem. We begin this section with some basic story planning definitions that enable the generation of example story plans and finish with an algorithm for constructing the IDG.

Definition 1 (Action) An action A consists of preconditions that must be satisfied before execution, $\textsc{Pre}(A)$, and effects that result, $\textsc{Eff}(A)$ (first and second columns, respectively, on right of Figure 2). Preconditions are literals in a state space whose conjunction must evaluate to true before execution. An action's effects are literals whose conjunction reflects the change in state space when A is executed.

Together with an action's name and parameter list, (first line of each square in Figure 2), the precondition and effects describe an *action schema*. An action schema can be instantiated into various forms, for example the *Move* action is executed by *rr*, *bw*, and *wm* in story-plan *A* from Figure 1.

Definition 2 (Story planning problem) A story planning problem Φ is a five-tuple $\langle \mathcal{I}, \mathcal{G}, C, S, \Lambda \rangle$ where $\mathcal{I}$ and $\mathcal{G}$ are conjunctions of True literals in the initial state and goal state respectively, C the set of symbols referring to character agents, S the set of symbols, and Λ a set of action schemata.

Examples of each element of Φ can be seen on the left of Figure 2. For illustration, we have explicitly separated character goals from the initial state. Typically they would appear using the *intends(character, goal)* literal, but we omit these details and the definition of an intentional plan for brevity, as they are not used by our algorithm. We do, however, define three character-centric structures from (Ware 2014) that both ensure all actions in a solution plan support character goals and for calculating efficient search strategies.

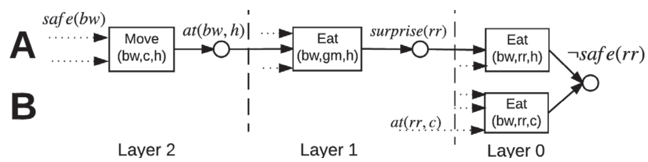

Figure 3: Goal graph with two paths for $\langle bw, \neg \mathit{safe}(rr) \rangle$. Here rectangles indicate steps with preconditions for those steps shown to the left of the rectangles and effects shown to the right.

Definition 3 (Character goal) A character goal is a tuple of two elements $\langle c, g \rangle$, where g is a literal that a character c desires to make true.

Character goals are accomplished through sequences of actions called intentional paths. Figure 3 illustrates two intentional paths for the character-goal $\langle bw, \neg \mathit{safe(rr)} \rangle$. Path A consists of three actions *bw* must take to accomplish $\neg$*safe(rr)*, while B is a single action. For brevity, the *surprise(rr)* predicate substitutes for more complex semantics allowing *bw* to surprise *rr*, as in the classic LRRH tale.

Definition 4 (Intentional path) An intentional path is constructed for $\langle c, g \rangle$ by causally linking c's actions until the final action has effect g. An action is only required to have one (or more) preconditions satisfied for it to be added to the path. The path must not contain a literal and its negation, nor can a literal appear twice.

Actions in an intentional path only require that at least one precondition to be satisfied through causal links to an earlier action's effects. Any remaining preconditions must then be satisfied outside of the intentional path, possibly from some other character's intentional path. This introduces an *intentional dependency*, one of two factors that fundamentally shape the possible story trajectories and is captured by the IDG. As an example from Figure 3, the *Move* action in layer 2 satisfies a precondition to the *Eat* action in layer 1, but *Eat*'s remaining preconditions (dotted incoming lines) must be satisfied by actions in service to some other character's goal. Note that intentional paths are constructed separately from the story plan graph.

Definition 5 (Causal links) A causal link, $s \xrightarrow{p} u$, is a tuple $\langle s, p, u \rangle$ where s, u are actions and p is a literal. A causal link records that p is both an effect of s and satisfies the precondition in u.

To improve planning efficiency, the Glaive narrative planner aggregates multiple intentional paths of a single character-goal into a character goal graph, represented in a layered graph (for a more in depth description of layered graphs in general see (Kivelä et al. 2014)).

Definition 6 (Character goal graph) A character goal graph, cgg, is a tuple $\langle CG, G \rangle$ where CG is a character goal and G a directed layered graph where vertices represent actions and layer 0 contains vertices that achieve the character goal. Action A appears in layer $i > 0$ iff A does not appear in a layer $i < 0$ and an effect e of $\textsc{Eff}(A)$ such that e is used as a precondition to an action B at layer $i-1$. An edge exists from A to B denoting $\textsc{Eff}(A)$ used in $\textsc{Pre}(B)$. A cgg set for Φ is $CGG(\Phi)$.

Character goal graphs enable an efficiency gain when constructing story planning graphs (Definition 8). They reduce the actions being considered to those which are *potentially motivated*. A character goal g is motivated when $\neg g$ is true.

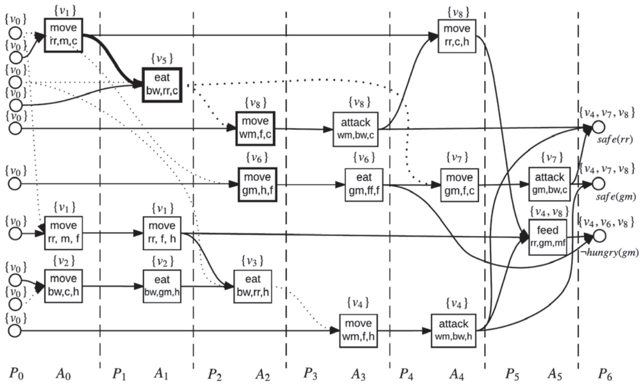

Figure 4: Simplified story planning graph (SPG) for LRRH. Here rectangles indicate steps with preconditions for those steps shown to the left of the rectangles and effects shown to the right

Definition 7 (Potentially motivated action) An action A of character c whose preconditions have been satisfied and is on at least one intentional path of a motivated character goal $\langle c, g \rangle$ is called potentially motivated. The potential motivations of an action is indicated by $\textsc{Mot}(A)$. The character goal $\langle bw, \neg \textit{safe}(rr) \rangle$ in Figure 3 is motivated because in the initial state, *rr* is *safe*.

Potentially motivated actions introduce the second of two factors that fundamentally shape the possible story trajectories: a *motivation dependency* between character goals, which we also represent in the IDG. For instance, when the *bw* accomplishes $\neg \textit{safe}(rr)$, it serves as a necessary condition before either the *wm* or *gm* adopts and acts in service of their $\textit{safe}(rr)$ goal. While story planning graphs are used for efficiency, they can also represent the individual intention and motivation dependencies of character goals.

Definition 8 (Story planning graph (SPG)) The SPG of a story planning problem Φ, $SPG(\Phi)$, is a directed layered graph. Each layer consists of two levels, literal P_i and action A_i, written $(P_0, A_0, P_1, A_1, ..., A_k, P_{k+1})$ for a k-layer graph. The levels P_0 and P_{k+1} contain the literals from $\mathcal{I}(\Phi)$ and $\mathcal{G}(\Phi)$, respectively, with *precondition* and *effect* edges connecting the layers in between (as in the original planning graph (Blum and Furst 1997)). An additional edge, a motivation edge, is used to indicate that for an action to be potentially motivated it requires the adoption of a new character goal. A motivation edge connects the literal $\neg g$ to an action at the highest layer of an intentional path (Definition 4) with goal g. Actions are added to the SPG when preconditions are satisfied and the action is *potentially motivated*.

The SPG in Figure 4 omits literal layers $P_1 - P_5$ and some actions for clarity. The bold *eat(bw,rr,c)* action in A_1 has an incoming precondition edge from *move(rr,m,c)* in A_0, indicating that character goal $\langle bw, \neg \textit{safe}(rr) \rangle$ has an an *intention dependency* on $\langle rr, \neg hungry(gm) \rangle$. The dashed outgoing motivation edges, $safe(rr)$, of *eat(bw,rr,c)* indicate that $\langle bw, \neg \textit{safe}(rr) \rangle$ fulfills a *motivation dependency* for both the *wm* and *gm* acting in service of their $\textit{safe}(rr)$ goal.

While individual character intention and motivation dependencies are represented in the SPG by precondition/effect/motivation edges, dependencies between character-goals are not explicitly represented. A primary contribution of this paper is the IDG, which aggregates the dependencies between character-goals from the SPG into exemplar trajectories. The final layer of vertices in the IDG are the exemplar trajectories, each a set of character-goal conjunctions. Together the exemplars estimate the story branches in the SPG to the narrative planning problem. From these exemplar trajectories, story plans can be generated.

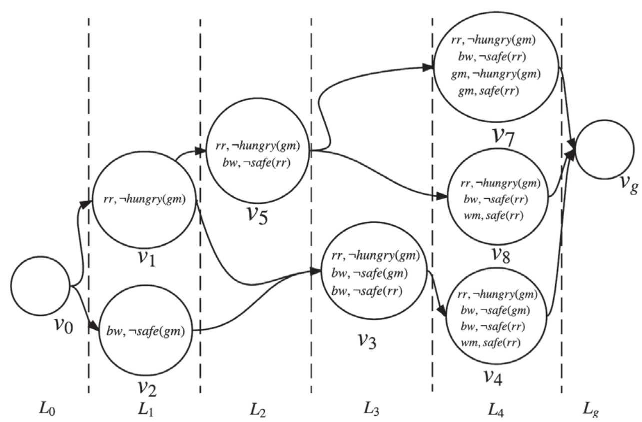

Figure 5: Intention dependency graph (IDG) for LRRH

Definition 9 (Intention dependency graph (IDG)) The IDG of a planning problem Φ, $IDG(\Phi)$, is a direct layered graph represented by a tuple of three elements $\langle V, E, f \rangle$ where V is a set of n vertices $v_0, v_1 \ldots, v_{n-2}, v_g$ each assigned a layer and a label, $L(v_i)$, consisting of a set of character goals. E is a set of directed edges between vertices, and f a vertex labeling function that takes a set of IDG vertices as input and outputs the union of their character goals.

Vertices are added to the IDG when an SPG action is the start of a new intentional path or when a new dependency is identified between existing intentional paths. An edge is added between IDG vertices to indicate a dependency (motivation or intention) in the character goals of the child vertex to those of its parents. The label of a vertex is constructed from the union of its parents' labels and new character goal (if the SPG action is the start of a new intentional path).

Our example continues with the IDG in Figure 5 constructed from the intention and motivation character dependencies of the SPG in Figure 4. Two vertices (v_1, v_2) are added to the IDG for the three potentially motivated actions in A_0 of the SPG, as they are the first actions towards two character goals. IDG edges are added between v_0 and v_1, v_2 indicating the intention and motivation dependencies to the initial state. The bold *eat* action in A_1 is the start of a new intentional path towards $\langle bw, \neg \mathit{safe}(rr)\rangle$, and a new vertex v_5 must be added to the IDG. To represent the intentional dependency (bold, solid edge) to the *move* action in A_0, we add an IDG edge from v_1 to v_5. The motivation dependency of $\langle bw, \neg \mathit{safe}(rr)\rangle$ to P_0 of the SPG is transitively satisfied in the IDG by v_5's connection to v_1, which depends on v_0.

The addition of IDG nodes and edges continues until we reach the goal-state in the SPG, at which point we connect v_g to IDG vertices whose character goals could satisfy the goal state. Our example shows v_4, v_7, and v_8 connected to v_g, with respective labels having intentional paths in the SPG. These three vertices are also representative of the three original trajectories in Figure 1 and are *exemplars* of the narrative content in solutions contained in the SPG. Furthermore, the IDG satisfies the *comparable* property by using character goals in the vertex labels. We can make comparisons between exemplar trajectories' character goals.

Finally, we address the *efficiency* property of an accommodation strategy by introducing *possible histories* to construct the IDG and SPG simultaneously.

Definition 10 (Possible history) A possible history (PH) is a set of IDG vertices $\{v_1, v_2, ...v_n\}$ associated with each SPG literal and action. The $PH(p)$, where p is a literal, captures the different ways p could become true from the possible history of actions with p as an effect. The $PH(A)$ of action A consists of PH constructed from union of the $PH(\text{PRE}(A))$ and $PH(\text{MOT}(A))$.

PHs are used to propagate IDG vertices in the SPG enabling new IDG vertices to connect to the character goals they depend on. We return to our example in Figure 4 and see IDG vertex v_0 is added to all initial state literals in layer P_0, denoted by $\{v_0\}$. The $move(rr, m, c)$ action in A_0 is dependent on the initial state, is potentially motivated by *rr*'s goal of $\neg hungry(gm)$, and is the first action of a new character goal. Since no precondition or motivation literals have a PH with this goal, we add a new IDG vertex with a label consisting of the union of the new character goal and the PH labels from its precondition literals. This results in v_5 being added to the IDG with the character goal $\langle rr, \neg hungry(gm)\rangle$ as a label, since v_0 is the only parent and has a blank label.

The simultaneous construction of the SPG and IDG is detailed in Algorithm 1. The initial setup, where v_0 is added both to the IDG and the PH of literals in P_0 of the SPG is detailed in lines 1-4. Lines 5-14 capture the end of SPG construction, when the goal literals of the planning problem have been achieved and the final IDG vertices are added, which represent conjunctions of character goals that estimate the possible solution story plans to the planning problem. In all other levels of the SPG (line 15), the algorithm enumerates the *possible histories* that could have lead to the action being executed and adds the action to the SPG (lines 18-21). If the action is part of an existing goal graph and no new dependencies are introduced, the action simply inherits the possible histories (line 26-27). Alternatively (lines 28-31), if the action introduces new dependencies, then a new vertex, v_k, is added to the IDG with a label containing the union of the character goals from IDG vertices it depends on. This is followed by adding an edge between the PH vertices and v_k in IDG and v_k being added to the possible histories of the action. Finally, if the action is part of a new character goal, then lines 33-38 add a new motivation edge to the SPG, add a new IDG vertex with the new goal, and connect this vertex to the IDG vertices it depends on. Lines 16-44 are repeated for each action added to the level and PH propagated to its effects, until no more actions can be added and we advance to the next SPG level (line 45).

We have addressed the three desired solution properties of accommodation by uncovering the underlying dependencies between character goals from the SPG. The IDG addresses the *exemplar* and *comparable* properties by representing the intentional and motivational dependencies between character goals as a conjunctive set that estimates the narrative content range of solutions in the SPG to the narrative planning problem. We address the *efficiency* property with an algorithm that constructs the SPG and IDG simultaneously.

Assessment

In our LRRH example, we obtain three IDG solution vertices $(v_4, v_7, v_8$, Figure 5) each with different character goals. These vertices are represented by the plans in Figure 1, where *A* is the classic tale, *B* closely resembles the classic tale but with *rr* immediately being eaten by the *bw* due to an exceptional user action. Finally, plan *C* differs a greatly from the classic tale, with *gm* fending for herself and assuming the hero role to rescue *rr*. We observe these differences numerically by applying the Jaccard distance, a typical method for plan comparisons (Srivastava, Nguyen, and Gerevini 2007), to the set of character goals in the labels of the aforementioned IDG vertices. If an EM accommodates by choosing to minimize the change in character goals from a desired trajectory (v_4) to a new one (v_7 or v_8), it would use a plan with character goals in $L(v_8)$. The choice would introduce a single character goal difference, the *bw* not eating *gm* and is reflected in a smaller Jaccard distance, 0.25 $(1 - \frac{3}{4})$, than v_7 (0.66 $(1 - \frac{2}{6})$). While using Jaccard distance of character goals is useful for our demonstration, it cannot account for all possible cases, such as those plans with different character goals but the same action set. Validating a narrative-theoretic distance metric is outside the scope of this work.

The IDG produces a set of exemplar trajectories. By maintaining all possible intentional and motivational dependencies between character goals as vertex labels, the IDG produces a set exemplar trajectories for $\mathcal{G}(\Phi)$. This disjunctive set of IDG vertices represents all possible conjunctions of character goals in the SPG and estimates the narrative content range of solutions to the story planning problem (Φ).

The IDG captures narrative-theoretic differences between solution plans. The primary narrative-centric addition that

Algorithm 1 $SPG(\Phi, CGG, level)$

Require: Story problem Φ, goal graphs CGG, level index
Ensure: SPG where final literal layer has $g_i \in \mathcal{G}$ true

If block for initial SPG and IDG setup
if $level == 0$ **then**
  Add v_0, v_g to IDG
  Add $\mathcal{I}(\Phi)$ to P_0 and Set $PH(\mathcal{I}(\Phi)) = v_0$ in SPG
  $SPG(\Phi, 1)$

Else If block to end SPG and IDG construction
else if $\forall g_i \in \mathcal{G}(\Phi) == \top \mid\mid P_{level-1} == P_{level}$ **then**
  for all vertex set $V \in PH(\mathcal{G}(\Phi))$ **do**
    if $|V| == 1$ **then** ▷ IDG vertex satisfies $\mathcal{G}(\Phi)$
      add edge $v \in V$ to v_g in IDG
    else ▷ new dependencies, new vertex in IDG
      create vertex $v_{|V|+1}$ with label $f(V)$ in IDG
      add edge from $v \in V$ to $v_{|V|+1}$, $v_{|V|+1}$ to v_g in IDG
    end if
  end for
  End

Else block for all other levels of the SPG
else
  for all ground action $A \in \Lambda(\Phi)$ **do**
    if A applicable at A_{level} & $A \in CGG$ **then**
      add A at A_{level} in SPG
      add edges from $Pre(A)$ in P_{level} to A in SPG
      add $Eff(A)$ at $P_{level+1}$ in SPG
      add effect edges from A to $Eff(A)$ in SPG
      for all char goal graph $cgg \in CGG(\Phi)$ **do**
        if $A \in cgg$ **then**
        For each IDG vertex set that makes A applicable
          for all vertex set $V \in PH(Pre(A))$ **do**
            if $CG(cgg) \in L(V)$ **then** ▷ existing goal in V
              if $|V| == 1$ **then** ▷ No new dependencies
                add $v \in V$ to $PH(A)$ in SPG
              else ▷ new dependencies, new vertex in IDG
                create $v_{|V|+1}$ with label $f(V)$ in IDG
                add edge from $v \in V$ to $v_{|V|+1}$ in IDG
                add $v_{|V|+1}$ to $PH(A)$ in SPG
              end if
            else ▷ new character goal, new vertex in IDG
              add motivation edge $\neg G(cgg)$ to A in SPG
              add $v_{|V|+1}$ with label $f(V)$ in IDG
              add edge from $\forall v \in V$ to $v_{|V|+1}$ in IDG
              add $v_{|V|+1}$ to $PH(A)$ in SPG
            end if
          end for
        end if
      end for
    end if
    add $PH(A)$ to $PH(Eff(A)$ in SPG ▷ propagate PH
  end for
  $SPG(\Phi, CGG, level++)$

end if

intentional planning makes to classical planning is ensuring character goals support authorial goals. The IDG identifies intentional and motivational dependencies between character goals from the SPG to create conjunctive sets of character goals to estimate the narrative content range in solutions to the story planning problem. Comparisons between IDG vertex labels captures their narrative-centric differences.

Constructing the IDG is more efficient than solving the planning problem. The SPG is equivalent to Glaive's planning graph and retains its increase in polynomial complexity over the original Graphplan planning graph. The Graphplan planning graph is constructed in polynomial time (Blum and Furst 1997), which is less than automated planning's P-SPACE complexity class (Bylander 1994). The simultaneous construction of the IDG adds three polynomial costs to Glaive's planning graph incurred on adding ground actions to the SPG. Firstly, we discuss IDG vertex additions. The maximum number of new IDG vertices per ground action can be computed from the product of its precondition and motivation possible history sizes, denoted as $v = |PH(\textsc{Mot}(A))| \times \prod_{i}^{|\textsc{Pre}(A)|} |PH(\textsc{Pre}_i(A))|$. In the worst case, every precondition and motivation to an action has all previous IDG vertices, so each combination of possible histories used to obtain an action's preconditions leads to a new IDG vertex and would lead to $|V(IDG)|^{|\textsc{Pre}(A)|+|\textsc{Mot}(A)|}$ vertices being added. Note the exponent is relative to the action's preconditions and motivations, and not the size of the IDG.

Secondly, there is a search cost to determine if an action's possible history already contains an action's character goal in the labels of IDG vertices. Since the labels are unordered, at worst this requires $c = |CGG(\Phi)|$ comparisons.

Finally, we calculate additional cost for IDG edge additions. For each new IDG vertex, it is possible to need a new edge for each of the ground action's preconditions and one for its motivation. We indicate this worst case scenario with $e = |\textsc{Pre}(A)| + 1$.

These three IDG costs are incurred per ground action at each layer of the SPG. The total cost of constructing the IDG is captured by $\mathcal{O}(knvce)$, where k is the number of levels in the SPG, n the maximum number of actions at any SPG level, and v,c,e as described in the preceding paragraphs.

The IDG has inherent limitations. While constructing the IDG addresses the desired properties of a plan-based accommodation solution, there is no free lunch. In our case, we trade-off speed for accuracy, as characterized by three factors. First, the SPG does not use *mutex* edges to capture precedence and interference relationships between actions. This results in unreachable plans being considered in the SPG and consequently overestimating the number of exemplars that could solve the narrative planning problem. Secondly, the SPG terminates construction when $\mathcal{G}(\Phi)$ is satisfied, capturing only solutions near minimal length. Non-optimal solutions with different character-goal dependencies may exist beyond where the SPG ends. Finally, only solution plans with character-goals necessary to support reaching $\mathcal{G}(\Phi)$ are considered. While compatible but *'dead-end'* branches (non-necessary) can add to a trajectory's narrative content, they are not considered by the SPG and IDG.

Conclusions and Future Work

An IN with a plan-based representation has the ability to represent branching stories, but suffers from high computational costs. These costs become particularly problematic when an EM must accommodate an exceptional user action and introduce a new narrative trajectory. We identified three desirable properties (exemplars, comparison, efficiency) that general solutions must address and detail how our plan-based representation, the IDG, addresses them.

The IDG first addresses the *exemplar* property by ensuring that a disjunctive set of IDG vertices represents all the possible conjunctions of character goals in the SPG to estimate the narrative content range of solutions (story branches) to the story planning problem. Secondly, comparing the labels of IDG vertices addresses the *comparison property* as the labels contain character goals, the primary addition story-plans make to classical plans. We follow classical planning's approach to comparing plans by using a Jaccard-based distance metric. Finally, the *efficiency* property is achieved by leveraging the reachability heuristic calculations done in polynomial time by Glaive's planning graph to simultaneously create the IDG. We demonstrate examples of how the IDG addresses these properties with a simplified LRRH domain.

An immediate area for continued work is to perform an empirical evaluation with two components. The first is will determine if the IDG generates a wide variety of solutions (within its own solution set). This can be measured using solution diversity (Coman and Muñoz-avila 2011) calculations that determine how effective the IDG is at finding different solutions. We expect the IDG would find more diverse solutions with less effort that existing approaches (e.g. (Ware and Young 2014)) as it leverages the underlying intention dependencies in the story planning problem. The second component to an evaluation is to determine if the IDG is generating the same solutions as other diverse story planners (e.g. (Farrell and Ware 2016)). This can be measured by using the solution overlap measure (Roberts, Howe, and Ray 2014). We expect that the IDG will find all the solutions the other story planners do, as well as others, should they exist in the solution space. Using both solution diversity and overlap against results from existing diverse planners would demonstrate the advantages of this paper's approach.

Acknowledgements

This material is based upon work supported in whole or in part with funding from the Laboratory for Analytic Sciences (LAS). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the LAS and/or any agency or entity of the United States Government.

References

Amos-Binks, A.; Roberts, D. L.; and Young, R. M. 2016. Summarizing and Comparing Story Plans. In *7th Workshop on Computational Models of Narrative (CMN 2016)*, volume 46, 107–123.

Bates, J. 1992. Virtual Reality, Art, and Entertainment. *Presence: Teleoperators and Virtual Environments* 1(1):133–138.

Blum, A. L., and Furst, M. L. 1997. Fast planning through planning graph analysis. *Artificial Intelligence* 90(1-2):281–300.

Bryce, D., and Kambhampati, S. 2007. A tutorial on planning graph based reachability heuristics. *AI Magazine* 28(1):47.

Bryce, D. 2014. Landmark-based plan distance measures for diverse planning. In *ICAPS*, 56–64.

Bylander, T. 1994. The computational complexity of propositional STRIPS planning. *Artificial Intelligence* 69:165–204.

Coman, A., and Muñoz-avila, H. 2011. Generating Diverse Plans Using Quantitative and Qualitative Plan Distance Metrics. In *AAAI Conference on Artificial Intelligence*, volume 18015, 946–951.

Farrell, R., and Ware, S. G. 2016. Fast and Diverse Narrative Planning through Novelty Pruning. *AIIDE*.

Helmert, M. 2006. The fast downward planning system. *Journal of Artificial Intelligence Research* 26:191–246.

Kivelä, M.; Arenas, A.; Barthelemy, M.; Gleeson, J. P.; Moreno, Y.; and Porter, M. A. 2014. Multilayer networks. *Journal of complex networks* 2(3):203–271.

Magerko, B. 2007. Evaluating Preemptive Story Direction in the Interactive Drama Architecture. *Journal of Game Development* 2(3):25–52.

Mateas, M., and Stern, A. 2005. Structuring content in the Façade interactive drama architecture. *Proceedings of the First Artificial Intelligence and Interactive Digital Entertainment Conference, June 1-5, 2005, Marina del Rey, California, USA* 3:93–98.

Nguyen, T. A.; Do, M.; Gerevini, A. E.; Serina, I.; Srivastava, B.; and Kambhampati, S. 2012. Generating Diverse Plans to Handle Unknown and Partially Known User Preferences. *Artificial Intelligence* 190:1–31.

Porteous, J.; Cavazza, M.; and Charles, F. 2010. Applying planning to interactive storytelling: Narrative control using state constraints. *ACM Transactions on Intelligent Systems and Technology (TIST)* 1(2):10.

Ramirez Sanabria, A., and Bulitko, V. 2014. Automated Planning and Player Modelling for Interactive Storytelling. *IEEE Transactions on Computational Intelligence and AI in Games* 7(4):375–386.

Riedl, M. O., and Bulitko, V. 2012. Interactive Narrative: An Intelligent Systems Approach. *AI Magazine* 34(1):67.

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

Roberts, D. L., and Isbell, C. L. 2008. A survey and qualitative analysis of recent advances in drama management.

International Transactions on Systems Science and Applications 4(1):61–75.

Roberts, M.; Howe, A. E.; and Ray, I. 2014. Evaluating diversity in classical planning. In *ICAPS*.

Robertson, J., and Young, R. M. 2015. Automated Gameplay Generation from Declarative World Representations. In *Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE-15)*, 72–78.

Srivastava, B.; Nguyen, T.; and Gerevini, A. 2007. Domain Independent Approaches for Finding Diverse Plans. In *International Joint Conference on Artificial Intelligence*, 2016–2022.

Telltale Games. 2012. The Walking Dead. [Game].

Ware, S. G., and Young, R. M. 2014. Glaive: A State-Space Narrative Planner Supporting Intentionality and Conflict. In *International Conference on Artificial Intelligence and Interactive Digital Entertainment*.

Ware, S. G.; Young, R. M.; Harrison, B.; and Roberts, D. L. 2014. A Computational Model of Plan-based Narrative Conflict at the Fabula Level. *IEEE Transactions on Computational Intelligence and AI in Games* 6(3):271–288.

Ware, S. G. 2014. *A Plan-Based Model of Conflict for Narrative Reasoning and Generation*. Phd dissertation, North Carolina State University.

Young, R., and Riedl, M. 2004. An architecture for integrating plan-based behavior generation with interactive game environments. *Journal of Game Development* 1(1):1–29.