

The Case for Intention Revision in Stories and Its Incorporation into IRIS — A Story-Based Planning System

Matthew William Fendt and R. Michael Young

Digital Games Research Center
Department of Computer Science
North Carolina State University
Raleigh, NC 27695

Abstract

Character intention revision is an essential component of stories, but it has yet to be incorporated into story generation systems. However, intentionality, one component of intention revision, has been explored in both narrative generation and logical formalisms. The IRIS system adopts the belief/desire/intention framework of intentionality from logical formalisms and combines it with preexisting concepts of intentionality in narrative. IRIS also introduces the crucial concept of intention revision for characters in the story. The intent of this synthesis is to create stories with dynamic and believable characters that update their beliefs, replan, and revise their intentions over the course of the story.

Introduction

Narratives are a fundamental way of expressing thoughts and communicating information. Narratives can be defined as "the recounting of one or more real or fictitious events communicated by one, two, or several narrators to one, two, or several narratees" (Prince 2003). A common form of narrative is a story. Stories are a special kind of narrative, as they encompass a set of characters and "the fictional reality in which the characters are living," (Rimmon-Kenan 2002) as well as usually being constructed to conform to some commonly accepted dramatic structure. To this day, stories are still almost exclusively hand-authored. Systems that automatically or procedurally generate stories are still in their infancy, largely due to the complications that creating "interesting" stories present. At best, the structure of an "interesting" story is complex, and at worst, the structure is still unknown. Narratologists like Propp, who analyzed the components of Russian folk tales (1968), have proposed common building blocks of narrative, but most problems in the field of story generation are still largely unsolved. One of the complications of computational story generation is the creation of believable portrayals of characters, their actions, and their intentions. One way to create these believable characters is to have them act in a coherent manner, performing only actions that stem from plausible motivations, updating their beliefs when appropriate, and replanning in the event of plan failure.

A way to model these believable characters is to represent their mental model using the cognitive be-

lief/desire/intention (BDI) framework (Bratman 1999). This captures 1) the character's beliefs, things it thinks are true about the world, 2) the character's desires, things it would like to make true in the world, and 3) the character's intentions, commitments to action. The BDI framework has many applications, such as capturing speech acts (Cohen and Levesque 1988), but when BDI is extended to include the concept of intention revision, it can be applied to narrative generation systems. We describe here one such system—Intention Revision in Storytelling system, or IRIS.

The IRIS planning system is intended to address the issue of character intentionality and intention revision in the context of story generation. Intention revision is the modification of a character's plan due to a change in its beliefs. IRIS is a hybrid centralized/emergent planner that provides rules for adopting and dropping intentions in the context of narrative and uses a BDI mental model for its characters.

The Need for Intention Revision in Stories

Intention revision is defined as the modification of a character's plan due to a change in its beliefs. Intention revision is important for stories for at least these five reasons:

1) **It keeps the characters involved in the story.** If a character quickly completes all of its intentions, it will not perform any more actions. The character will effectively be eliminated from the story. However, the character can be prevented from completing at least one of its intentions. This allows for the possibility that another character will thwart the character's intentions and force it to replan, extending its plan and keeping the character involved in the story.

2) **It contributes to believable and interesting stories.** Without intention revision, the character will have the same set of intentions at the beginning of the story as at the end of the story, which prevents the kind of character growth common to interesting stories. If the character changes its intentions over the course of the story, the reader has a chance to learn along with the character.

3) **Conflict necessitates it.** We say that a story contains a conflict when one or more characters want two or more mutually exclusive conditions. At most, one character can win the conflict. For the characters that fail, they need to find another plan to fulfill their intentions. However, it is possible that there is no other plan that does so. In which case, the characters should drop the impossible intentions, and perhaps adopt new intentions.

4) **It allows for more complex character personalities.** Characters that have static intentions are less interesting than characters with dynamic or changing intentions. Over the course of the story, the character's personality may change, and it may work to achieve intentions that would have seemed at odds with the character at the beginning of the story.

5) **It allows for the assimilation of new information.** If a character receives an update in its beliefs, the character should incorporate this new information into its plans. This may cause the character to change its plan, or even to add or drop some of its intentions. Without intention revision, the character will be forced to ignore any new information and stoically work towards the intentions that the character formed at the beginning of the story.

It seems clear from the reasons above that a robust computational story generator should include the ability for characters to perform intention revision. However, intention revision is not yet a common feature in these generation systems. Some systems do include an important component of intention revision discussed above: intentionality. Intentionality is the way that the characters can influence story action selection, since the system's action selector will choose actions that help the character achieve its goals. Without intentionality, there is no narrative incentive for the generation system to select one character to perform an action in the story over any other.

Related Work

One of the foundational projects in story generation is TALE-SPIN, a fable generator (Meehan 1977). TALE-SPIN produces plans that incorporate story elements such as characters' bodily needs, achievement goals, and preservation goals. There have been many examples of story planners since TALE-SPIN that have emphasized different aspects of story. Sgouros' Plot Manager (1999) recursively generates new steps to a story while there is still something "interesting" to be added to the story, while using a series of favorable and unfavorable outcomes to help guide the story creation. Cavazza et. al produced the I-Storytelling System (2002), which uses an interactive, hierarchical task network- based personality approach to produce stories in the Source engine. Rizzo et. al (1999) also designed a heavily personality- based system. One of the disadvantages of these systems is the lack of intentionality from the characters. Without intentionality, it is difficult to show proper motivation for the characters' choice of actions.

Both Bratman (1990) and Pollack (1990) discuss intentionality in the context of planning, though not necessarily with regards to narrative. IPOCL (Riedl and Young 2004) also incorporates this intentionality into story generation. However, in IPOCL there is not the ability for a character to begin a plan, abandon it partway, and revise its intentions. In this paper, we propose a planning system targeted at story generation that addresses intentionality, in particular, intention revision. The proposed approach uses a BDI framework that models intentionality motivated by work such as Cohen and Levesque (1990) and van der Hoek (2007).

Beliefs, Desires, Intentions

The three components of the BDI framework are beliefs, desires, and intentions. First, beliefs are that which the agent believes to be true. They may or may not reflect the actual state of the world. Indeed, they often do not, since the agents may believe things that are not true. In the case that they do not match the true state of the world, it is possible that the agent will run into trouble. Failure could also result due to another agent's interference. Second, a desire is a world state that an agent would ideally like to be true. However, a set of desires do not have to be consistent. Additionally, desires do not have to be consistent with the agent's beliefs (van der Hoek, Jamroga, and Wooldridge 2007). For example, an agent could have the desire to become a millionaire, but not believe that this desire is possible. Finally, intentions are the desires that the agent chooses to act upon. Not all desires need be converted to intentions at some point in time. An agent could desire to have a cake, but never form the intention to bring this about by either baking the cake or purchasing it. An important aspect of intentions is that while beliefs and desires are usually in the form of propositions, intentions imply some commitment on behalf of the agent (Cohen and Levesque 1990).

The IRIS System

The Intention Revision in Storytelling (IRIS) system is a planning system whose purpose is to generate stories using a BDI framework with special regard to characters' intentions and intention revision. Each character will form its own individual plan to satisfy its intentions using a partial order planner (POP). IRIS does not use interleaved planning and execution, but rather constructs dynamic character plans during the algorithm's planning stage. When the character receives a belief update during the course of story generation, it may perform intention revision, altering its current plan or intentions. As for story creation, the character plans are combined into a single story by a Drama Manager (Kelso, Weyhrauch, and Bates 1992). This combination will continue until all of the authorial goals have been satisfied.

The IRIS Story Generation Algorithm

The IRIS story generation algorithm combines elements of both emergent and directed story generation. It uses individual character planning to generate a plan for each individual character and a Drama Manager to integrate each character plan into a centralized story plan. The advantage of using this hybrid approach over a centralized planner is twofold. First, the hybrid approach addresses the need to have per character plans and character specific knowledge. Second, the hybrid approach better lends itself towards intention revision. In the hybrid approach, each character has its specific plan and the Drama Manager has global knowledge of the entire planning domain. When an individual character plan fails, it is easy for that character to generate a new plan, the Drama Manager to receive that new plan, and the story plan integration to continue.

Though each character generates its own plan, a Drama Manager selects from among the next possible actions available in the characters' plans. This allows for a degree

Algorithm 1 IRIS-Narrative-Generation

Algorithm 1: IRIS-Narrative-Generation(C, G_A, Λ)
Initialization: Let $G_{open} = G_A$, $P = \emptyset$, $G_{closed} = \emptyset$, $G_{next} = \emptyset$. For each $c \in C$, $G_c = \text{partition}(G_A, c)$, $I_c = \text{create-intentions}(D_c, G_c)$, $P_c = \text{find-plan}(c, B_c, I_c, \Lambda)$
1. **Termination:** If $G_{open} = \emptyset$, report success and return P .
2. **Authorial goal selection:** If $G_{next} = \text{null}$, choose $G_{next} \in G_{open}$
3. **Character action selection:** Choose $c \in C$ with plan P_c . Let action $a = \text{first}(P_c)$. Let $P_c = \text{successor}(P_c)$. Let $P = P + a$. If (G_{next} in effects of a), $G_{closed} = G_{closed} + G_{next}$, $G_{open} = G_{open} - G_{next}$, $G_{next} = \text{null}$.
4. **Belief update:** For every $c' \in C$, let $B_{c'} = B_{c'} + \text{effects of action } a$.
5. **Intention revision:** Let $G_{c'}$, $P_{c'}$ be updated based on intention revision.
6. **Check for authorial goal failure:** If G_{next} is not possible to satisfy, $G_{next} = \emptyset$.
7. **Recursive invocation:** Call IRIS-Narrative-Generation(C, G_{open}, Λ)

of author-centric control that is not an option in emergent systems. As the characters gain new information and fail in their plans, they revise their intentions and replan. This allows for the system to produce characters with realistic behavior.

Definition: A *BELIEF* is a literal, and a *DESIRE* is a $\langle \text{literal}, \text{weight}, \{A, M\} \rangle$ triple, where the weight is an integer such that $0 < \text{weight} \leq \infty$, A represents that the desire will be converted to an achievement goal, and M represents that the desire will be converted to a maintenance goal.

Definition: An *INTENTION* is a belief that the character has committed to act upon.

Definition: A *PERSONALITY* is a string description of a character's personality that is used in the "adding intentions" phase of intention revision discussed below.

Definition: A *CHARACTER* c is a 7-tuple $\langle N_c, E_c, B_c, D_c, I_c, \Lambda, P_c \rangle$ where N_c is the character's name, E_c is the character's personality, B_c is the character's set of beliefs, D_c is the character's set of desires, I_c is the character's set of intentions, Λ is the set of actions the character can perform, and P_c is a totally ordered set of actions in Λ that when executed, satisfy the intentions in I_c .

Definition: *AUTHORIAL GOALS* G_A are a pair $\langle D_A, i_A \rangle$, where D_A is a set of desires denoting the conditions in the world the author wants to hold in the goal state, and $\forall d \in D_A, d_{\text{weight}} = \infty$. i_A is a nonempty set of characters such that $\forall c \in i_A, c \in C$.

Algorithm 1 describes IRIS's narrative generation process. The input for the algorithm is a set of characters C , a set of authorial goals G_A , and a set of actions Λ . The narrative generator first takes in the initial information about the story. Each character has a set of beliefs and desires. The authorial goals are assigned to the characters and the characters form their maximally-weighted consistent subset of

intentions from their desires and assigned authorial goals. The characters now have a set of beliefs as a starting state and intentions for an ending state, so they use these to form their initial plans.

If there are no more authorial goals to satisfy, the algorithm returns the completed plan. Otherwise, if the Drama Manager needs a new authorial goal to work towards, it selects one, as described in Algorithm 2.

The Drama Manager then selects the character action that will be chosen to be added to the story. The process through which this selection is done is described in Algorithm 3. The chosen action a is the first open action in the chosen character's plan. The action is removed from the character's plan and added to the story plan. If the authorial goal was one of the effects of the action, it is added to the closed authorial goals, removed from the open authorial goals, and the next authorial goal to satisfy is set to null.

All characters update their beliefs based on the effects of the chosen action (see Planning and Intention Revision in Story Generation Section), then have the opportunity to perform intention revision (see Intention Revision Section).

It is possible that the current authorial goal can no longer be satisfied. If that is so, then it is set to null. Then, the algorithm is recursively called.

IRIS's Belief/Desire/Intention Framework (Initialization)

In IRIS, each character in the story has its own BDI mental model. At the beginning of the story generation algorithm, each character selects a consistent set of desires to convert to its intentions. These desires are weighted. Additionally, the author provides a list of authorial goals that must be satisfied in the story and which characters can be selected to complete them. For example, $\langle \text{dead}(\text{DRAGON}), \{ \text{KNIGHT}, \text{MAIDEN} \} \rangle$ indicates that the (dead DRAGON) goal can be given to either the KNIGHT or the MAIDEN. The recipient of this authorial goal is chosen nondeterministically. The motivation behind these authorial goals is to allow the author to have certain goals be fulfilled in the story that no character necessarily wants to do. Maybe neither the knight nor the maiden have a desire to kill the dragon, but the author wants that goal satisfied. When one of the two characters is assigned the goal, it will form plans that will fulfill the author's goals.

After the authorial goals are distributed, the character selects its maximally weighted consistent subset of desires that are also consistent with its given authorial goals. The authorial goals have a weight of ∞ , which has two consequences. First, it ensures that all of the authorial goals will always be converted to the character's intentions at this step. Second, when the character performs intention revision and possibly drops some of its intentions, it will never drop an authorial goal until it is completed. This guarantees that at any step of Algorithm 1's execution, all of the unsatisfied authorial goals will be located in some character's set of intentions. This has useful properties that will be discussed later.

The authorial goals are always given to the characters as maintenance goals. A maintenance goal is a goal that the character will retain in its intentions and always plan to keep

Algorithm 2 Choose-Next-Authorial-Goal

Algorithm 2: Choose-Next-Authorial-Goal (C, G_{open})**Initialization:** $G_{available} = \emptyset$

1. **Find available authorial goals:** For each $c \in C$, find all $g \in G_{open}$ such that g is an effect of an uncompleted action $a \in P_c$ and no other g has yet been found in P_c . Add these g 's to $G_{available}$.
 2. **Find goals with fewest number of conflicts:** Remove all $g \in G_{available}$ such that g does not have the fewest number of conflicts in $G_{available}$.
 3. **Find goals that are not effects of the last action in its character's plan:** As long as there are goals $g \in G_{available}$ that are not completed in the last action of a character's plan P_c , remove all $g \in G_{available}$ such that g is an effect of action $a \in P_c$ and a is the last element of P_c .
-

true. The opposite of a maintenance goal is an achievement goal, which will be dropped from the character's intentions after it has been satisfied. The reason authorial goals are always maintenance goals is that it does not make narrative sense for a character to satisfy a goal that the author wants to keep true for the duration of a story and then for the character to adopt a plan that would undo the authorial goal.

Once the character has established its intentions, it will call a partial order planner to form its plan. The character's beliefs will serve as the initial state for the planner's input, and the intentions as the final or ending state. The resulting plan will allow the character to achieve its intentions if the character's beliefs match the world conditions throughout the execution of the plan. The plan is not guaranteed to succeed, however, since this plan will be given, along with other (potentially conflicting) character plans, to the Drama Manager, which will select actions from among these plans to be added to the story. In fact, it is desirable to introduce conflict so that the characters' plans fail and they have to perform intention revision. We suggest that this dynamic character behavior is essential to creating interesting stories.

Selecting Authorial Goals (Step 2)

The authorial goals are the author-supplied, minimal ending constraints of the story. Once satisfied, the authorial goals will not be reversed.

The challenge in selecting authorial goals is that we want to simulate the typical narrative structure of rising action, climax, and resolution. However, we do not want to have the author order the authorial goals by increasing dramatic value. This is because such an ordering will be essentially providing an outline to IRIS, decreasing IRIS's generative power. So, we want to potentially generate any possible dramatic ordering of the authorial goals such that the authorial goals are chosen with increasing dramatic value. We will measure dramatic value by the number of actions that could cause intention revision on the action that satisfies the authorial goal. Algorithm 2 selects the next available authorial goal with the fewest number of these intention revisions as long as the action that fulfills that goal is not the last action

D	M	K
1	4*	5
2		6*
3		7*
		8
		9
		10*

Figure 1: Example character plans for the "Selecting Authorial Goals" step of the IRIS algorithm.

Algorithm 3 Choose-Next-Character-Action

Algorithm 3: Choose-Next-Character-Action ($C, G_{open}, G_{closed}, G_{next}$)**Initialization:** $A_{available} = A_{initial} = \emptyset$

1. **Find available actions:** For each $c \in C$, $a \in P_c$, add a to $A_{available}$ if a is an uncompleted action and no effect of $a \in G_{open}$. $A_{initial} = A_{available}$.
 2. **Remove last actions:** For each $a \in A_{available}$, remove a if a is the last action in its character's plan P_c unless an effect of $a = G_{next}$.
 3. **Remove any reversal actions:** For each $a \in A_{available}$, remove a if an effect of $a = \sim g_{closed}$, where $g_{closed} \in G_{closed}$.
 4. **Removed revenged-upon actions:** For each $a \in A_{available}$, remove a if a is "revenged-upon" and $a' \in A_{available}$ such that a' is "revenge."
 5. **Selected from available action:** If $A_{available} \neq \emptyset$, nondeterministically return an $a \in A_{available}$.
 6. **Selected a failure action:** If $A_{available} = \emptyset$, nondeterministically select an $a \in A_{initial}$. Set a to "failed." Return a .
-

in that character's plan. This will cause the authorial goals to be selected in increasing dramatic value. Also, by avoiding selecting the last action in a character's plan, we have an opportunity to keep that character involved in the story longer. This is because a future revision of that character's plan might extend its plans and give it more actions to complete.

For example of authorial goal selection, refer to Figure 1. Each column represents a character's plan. The numbers are actions, and the actions are ordered with the earlier actions above later actions. The starred actions fulfill authorial goals. If we were selecting the next authorial goal to satisfy, the choices would be 4 and 6 (7 and 10 are not available because they come later in the K character's plan). Perhaps completing action 2 will cause action 4 to fail, but none of actions 1-6 will cause action 6 to fail. The goal satisfied in action 6 will be marked as the next authorial goal to satisfy.

Selecting the Next Character Action (Step 3)

Once an authorial goal has been selected to be fulfilled, there are a number of actions that can be chosen before the actual action that satisfies the authorial goal. The actions are chosen in such a way that both enhances narrativity and prevents the planner from failing to generate a complete story.

D	M	K
15	4*	8
16		9
1		10*
2		11
3		

Figure 2: Example character plans for the “Selecting Next Character Action” step of the IRIS algorithm.

The initial pool of actions is all of the topmost uncompleted character actions that do not satisfy a new authorial goal. For example, refer to Figure 2. If action 10 has the authorial goal we are trying to satisfy, then the initial pool of actions is (15, 8). The entire set of possible actions is (15, 16, 1, 2, 3, 8, 9, 10). However, an action later in a character’s plan is not available for selection until all preceding actions are selected. If action 15 is chosen, then 16 will then be available for selection, and so on. Note that 4 is not a possible choice because it will satisfy a new goal. All characters’ last actions are removed from the available pool for the same reasons described previously. In our example, this would eliminate action 3. Also, once fulfilled, no authorial goal can be reversed. So, all available actions that reverse any authorial goal are removed. Action 16 might reverse the authorial goal in 10, so it is removed. Then, if there are any “revenge” actions (discussed later), they will come before the action that satisfied the “revenged-upon” intention. This allows the vengeful characters to take their revenge and force the victim to perform intention revision. In the example, action 15 is a revenge intention, but the action that satisfies the revenged-upon intention is not an available action, so no actions need to be removed from the available action pool. This leaves actions 15 and 8 initially available for selection, and action 9 available after 8 is selected, and 10 after 9. Action 15 or 8 is chosen nondeterministically, and IRIS is ready for the next action to be selected. Once the action that satisfies the current authorial goal is chosen, IRIS selects a new authorial goal (See Selecting Authorial Goal Section).

It is possible that there are no available actions left by step 7 of Algorithm 3. If this is the case, one of the actions found in step 2 is selected. However, the Drama Manager will cause this action to fail. Once the action fails, characters may end up performing intention revision based on this failure, resulting in plans with valid actions.

Belief Update (Step 4)

Belief update is the primary catalyst that can cause an intention revision. A character can receive a belief update from one of two different sources: from the effects of an action after an action has been performed or knowledge acquisition from another character. The first source, from the effects of an action, is checked during the story generation process after a character’s action has been added to the story. Every character gets updated with the effects of the chosen action. There is a concern with this universal belief update, in that characters may gain new knowledge based on the effect of an action that they were not present to witness. However,

limiting the belief update to only the characters that were present for the action has problems as well. For example, character A and character B are at different locations. A originally knows the location of B. B moves from its original location. Since A was not present for B’s move action, it does not know where B is. By the nature of how planning works, A will not be able to form any plans that requires it to know the location of B.

A character’s beliefs can also be updated by other characters during story generation. This implies that a character can have the intention to change the beliefs of another character. The inform action would look something like

inform(A B belief)

where A is the character passing along the information, B is the character receiving the information, and belief is the belief being conveyed. By using the inform action, a character can instruct another character or even pass along false information for dramatic purposes. To give characters some resistance to false information, the inform action will only succeed if the informee trusts the informer. For example, a villain might want to turn Bob against his friend Joe by providing false information about Joe to Bob. If Bob learns this information, he will change his plan, which may have originally including collaborating with Joe. Bob may then refuse to work with Joe. To accomplish this nefarious goal, the villain first needs to perform some actions to gain the trust of Bob. When Bob trusts the villain, the inform action can successfully modify Bob’s beliefs.

The trust world condition and inform action are simplified versions of inter-agent trust and speech act models, respectively. These expanded models could be substituted for trust and inform, but they are not necessary since they each have their own unique purposes: collaborative action and refined agent communication. Since IRIS is using the inform action in service of story generation with attention to intention revision, this simplified version of agent communication is sufficient.

Planning and Intention Revision in Story Generation (Step 5)

Planning IRIS’s function is to produce interesting stories where characters’ plans are frequently in conflict. IRIS uses the partial order planning (POP) algorithm called Longbow (Young 1994). The entirety of the IRIS story generation is all done off-line and not in real time. There is not a sense of mixing planning and execution. Even the intention revision that takes place over the course of story generation is still done at planning time. The intention revision occurs during story generation when the character can no longer adhere to its plan and/or intentions that it held previously in the generation process.

Plan Failure A few words on plan failure are necessary. Plan failure occurs when a character’s plan will no longer achieve its intentions. This occurs when the character has an erroneous set of beliefs. The general form a plan failure takes is:

Believes(t, p) \wedge \neg Holds(t,p) \wedge Happens(t,A) \wedge Precond(A,p)

where t is a certain time in the story, p is a world condition, and A is an action. It is also worth noting that a character's beliefs can be incorrect independent of any other character's involvement. Thus, a character's plan can fail in this manner even if there are no other characters in the story. Finally, the character may still continue perform actions in a doomed plan. This is because the character is not omniscient and would automatically know as soon as the plan has failed. Only once the character receives a belief update that indicates the failure of the plan will the character perform intention revision.

Intention Revision Intention revision is a series of steps that a character will perform when it abandons its original plan so that it can adopt a new plan. This may include the character abandoning some of its intentions and/or adopting new intentions. Intention revision is not to be confused with plan revision. Plan revision is an alteration in the character's plan without modification of its intentions. Plan revision is an aspect of intention revision, though. As for the components of intention revision, they are:

1) Select another available plan that fulfills the character's intentions. The Longbow POP algorithm used to find the character's plan can return multiple plans. By default, the character adopts the highest ranked plan according to whatever heuristic is given to the planner. However, if this "best" plan fails, the character can replace its old plan with the next best plan that will still achieve its intentions. There are certain situations where this would be helpful. For example, if the character's plan has failed, but it has not received any useful belief update, replanning will not yield a more useful series of plans. Selecting the next best plan may result in a plan that will allow the character to fulfill its intentions. However, if there are no more available plans left for the character to try, it will proceed to part 3. Better yet, if it receives a helpful belief update, it will go to part 2.

2) Replan, using the character's new set of beliefs. If the character receives a belief update, it has a better option than trying plans based on old knowledge like it did in part 1. Instead, the character will first perform a check to see if the new information invalidates its current plan. If it does, the character will run its POP algorithm again with the updated beliefs. With the new information, the character will abandon its old plan and adopt a revised one based on the character's updated belief about the world. If the belief update is from false information (whether from a flawed observation or misinformation from another character), the character could actually abandon a legitimate plan and adopt a faulty one. This is not a problem, however, since the characters will still act according to their beliefs and fail in a dramatic fashion.

3) Adopt replacement intentions. Drop intentions until a working plan can be found. If no plan will result in the fulfillment of the character's intentions, the Drama Manager may cause the character to adopt replacement intentions. The reason that new intentions are added before any intentions are dropped is because it is possible that the new intentions that are adopted may be incompatible with some of the character's old intentions. If the intentions were dropped before the replacement intentions are added, then the character

may immediately have a plan failure. There are a few ways new intentions can be adopted:

a) Personality-based behavior. The characters in the story can be given personalities that can influence the frequency and type of intentions the drop and adopt. For example, stubborn personalities may be unwilling to drop intentions even after the intentions seem to be impossible to achieve, where as meek personalities may drop an intention after the first failure to achieve it. Similarly, characters can have personalities that change the type of intentions they adopt, such as showy or humble (see part c). These personalities could be mutable based on the frequency and type of intention revision the character performs over the course of the story. Even a stubborn character may have a change of heart in the face of repeated failure.

b) Adopt revenge intentions. If a character A has its intention thwarted by another character B and A is of vengeful personality, A can adopt the negation of one of B 's maintenance intentions as one of its own intentions. A maintenance intention is an intention that a character will attempt to keep satisfied at every step in its plan. This is opposed to an achievement intention which needs only be satisfied once. As an example of a revenge intention, perhaps B took an action that causes A to drop the achievement intention $have(A\ boat)$. B also has a maintenance intention $beAlive(B)$. A can adopt the intention $\sim beAlive(B)$. A 's actions that lead up to this goal would also be marked as revenge actions. This comes into play when the Drama Manager decides which character's action to select to add to the story (See Selecting the Next Character Action Section).

The way revenge intentions are considered here is a simplification in two ways. First, we are only considering revenge intentions. There exist a range of other narrative features, such as deception, competition, avoidance, etc. Revenge intentions are considered because the adoption of these intentions is most directly formalizable in terms of intention revision. A more general approach to adopting replacement intentions would be able to incorporate more of these narrative features. However, there is no need for special modifications to the BDI framework when these features are added, since they can be represented with the existing BDI structures. The second simplification comes from the way revenge intentions are considered. The notion of revenge intentions can be expanded and generalized.

c) Adopt a similar intention. The story author can provide some descriptors that are associated with the possible story intentions. For example, $(haveYacht)$ might be associated with $(rich, attention_seeking)$. Based on the character's personality, if it drops an intention, the can adopt a similar intention as a replacement. This simulates a character shifting its efforts to a related or second-best goal that is still consistent with its personality. A showy character that fails to achieve $(haveLimo)$ may adopt $(haveYacht)$, but a more humble character that wanted the limo solely as a source of transportation may not.

After replacement intentions have been adopted, the character will drop a set of minimally weighted intentions until it can find a plan that it believes will succeed. As discussed above, the character will never drop an authorial goal this

way.

These three components of intention revision allow the character to continually generate plans that allow it to work towards the character's intentions. Intention revision allows the character to assimilate new information and to modify its intentions based on these belief updates. This ensures that the character stays active throughout the course of the story. It will not simply fail to achieve its intentions and drop out of the story.

Checking for Authorial Goal Failure (Step 6)

Once all of the characters have had a chance to perform intention revision, it is possible that character plans have changed such that the action that satisfies the current authorial now comes after some other authorial goal. For example, the current authorial goal is satisfied in action 8 and a character has the plan (8*, 9*, 10). After replanning, the character now has the plan (9*, 8*, 10). It is not possible to satisfy the current authorial goal without satisfying the goal in action 9 first. IRIS will need to select a new authorial goal to work towards. It is possible that IRIS will immediately select the goal in action 9 and then the goal in action 8, but it is also possible that some other authorial goals will be satisfied before the goal in action 8 is attempted again. The reason for dropping authorial goals is to achieve the rising action framework previously discussed.

IRIS's Generative Power

IRIS is a sound and complete narrative generation system. IRIS takes initial information from the author and ceases generation when either all authorial goals are satisfied or it is impossible to satisfy all of the authorial goals. Algorithm 1 will not return a complete story if all of the authorial goals are not satisfied. IRIS is also complete when backtracking is added. Due to the nondeterministic nature of authorial goal assignment and character action selection, the Drama Manager could make a valid selection that will later prevent the completion of the authorial goals. However, if the system fails to find a complete plan, it will backtrack to each choice point and resume generation from that point. Therefore, IRIS is complete since if there is a valid complete story that can be generated with the initial information, then that story will be generated.

One of the challenges of using planning to perform story generation is that typical planning approaches do not address interestingness. Soundness and completeness of a planner do not get far in terms of interestingness of the stories generated, so we offer one contribution to creating interesting stories in the form of intention revision. However, we acknowledge that intention revision is just one component that a robust story generation system should contain.

Conclusion and Future Work

The IRIS system is intended to introduce intentionality and intention revision into story generation. The characters in the story generation process harness a belief/desire/intention framework to generate plans that are designed to fulfill their intentions. These individual character plans are then woven

into a unified story by a Drama Manager. The Drama Manager fulfills the authorial goals in increasing dramatic value. Also, a character is not forever locked into its initial plan. Whenever a character's beliefs are updated, the character can perform intention revision. This allows for dynamic and realistic character behavior in the story.

There is still much to explore in the context of intentionality and intention revision in story generation. The concept of authorial goal assignment might be expanded to give the author control over story conditions other than ending conditions. For example, the author might be allowed to force a character to use a certain action or actions to achieve the character's goals. Additionally, an experiment will be conducted using the QUEST question answering system (Graesser, Lang, and Roberts 1991) where the stories generated by IRIS will be shown to users and the users will determine if they demonstrate dynamic and believable character behavior.

References

- Bratman, M. 1990. What is intention. *Intentions in communication* 15–31.
- Bratman, M. 1999. Intention, plans, and practical reason.
- Cavazza, M.; Charles, F.; and Mead, S. 2002. Planning characters behaviour in interactive storytelling. *The Journal of Visualization and Computer Animation* 13(2):121–131.
- Cohen, P., and Levesque, H. 1988. Rational interaction as the basis for communication.
- Cohen, P., and Levesque, H. 1990. Intention is choice with commitment. *Artificial intelligence* 42(2-3):213–261.
- Graesser, A.; Lang, K.; and Roberts, R. 1991. Question answering in the context of stories. *Journal of Experimental Psychology: General* 120(3):254–277.
- Kelso, M.; Weyhrauch, P.; and Bates, J. 1992. *Dramatic presence*. Citeseer.
- Meehan, J. 1977. Tale-spin, an interactive program that writes stories. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*.
- Pollack, M. 1990. Plans as Complex Mental Attitudes Martha E. Pollack. *Intentions in communication* 77.
- Prince, G. 2003. *A dictionary of narratology*. Univ of Nebraska Pr.
- Propp, V. 1968. *Morphology of the Folktale*. Univ of Texas Pr.
- Riedl, M., and Young, R. 2004. An intent-driven planner for multi-agent story generation. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 1*, 186–193. IEEE Computer Society Washington, DC, USA.
- Rimmon-Kenan, S. 2002. *Narrative fiction: Contemporary poetics*. Routledge.
- Rizzo, P. 1999. Goal-based personalities and social behaviors in believable agents. *Applied Artificial Intelligence* 13(3):239–271.
- Sgouros, N. 1999. Dynamic generation, management and resolution of interactive plots. *Artificial Intelligence* 107(1):29–62.
- van der Hoek, W.; Jamroga, W.; and Wooldridge, M. 2007. Towards a theory of intention revision. *Synthese* 155(2):265–290.
- Young, M. 1994. A developer's guide to the Longbow discourse planning system. *University of Pittsburgh Intelligent Systems Program Technical Report* 94–4.