ABSTRACT

FENDT, MATTHEW WILLIAM. Leveraging Intention Revision in Narrative Planning to Create Suspenseful Stories. (Under the direction of Dr. R. Michael Young.)

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 the protagonist of the story. The IRIS system uses its generative power alongside psychological and narrative models of suspense to computationally create suspenseful stories.
Leveraging Intention Revision in Narrative Planning to Create Suspenseful Stories

by

Matthew William Fendt

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Computer Science

Raleigh, North Carolina

2014

APPROVED BY:

Dr. David Roberts

Dr. Robert St. Amant

Timothy Buie

Dr. R. Michael Young
Chair of Advisory Committee
Matthew Fendt grew up in Aston, PA. In 2009 he received a B.S. in Computer Science from University of Delaware, graduated cum laude, and was awarded the Computer and Information Science Outstanding Student Award, awarded to the top Computer Science student. Matthew performed research for the first time at University of Delaware. He enrolled as a Ph.D. student at North Carolina State University and continued pursuing research with the Liquid Narrative research group. He is now a Lecturer of Computer Science at Baylor University.
ACKNOWLEDGEMENTS

I would like to thank my advisor, R. Michael Young, and my committee: David Roberts, Robert St. Amant, and Timothy Buie. I would like to thank the National Science Foundation for providing the grant # IIS-0915598 which helped fund my research. Thank you to my fellow lab mates who helped with the discussion and exchange of ideas in my research. Finally I would like to thank my family for their support.
# TABLE OF CONTENTS

LIST OF TABLES ........................................ vi

LIST OF FIGURES ........................................ vii

Chapter 1 Introduction .................................... 1
  1.1 Introduction ........................................ 1
  1.2 Motivation ........................................... 2
  1.3 IRIS Story Generation Process ......................... 4
  1.4 Contributions of the IRIS System ...................... 4

Chapter 2 Related Work ................................... 6
  2.1 Introduction .......................................... 6
  2.2 Understanding Narrative ............................... 7
  2.3 Suspense in Narrative ................................ 8
  2.4 Evolution of Narrative Generation ..................... 10
  2.5 Planning .............................................. 10
    2.5.1 Motivation ....................................... 10
    2.5.2 Description ....................................... 10
  2.6 Beliefs, Desires, and Intentions ....................... 12
    2.6.1 Motivation ....................................... 12
    2.6.2 Description ....................................... 13
    2.6.3 Cohen and Levesque’s Formalism of Intentionality .. 13
    2.6.4 Logical Formalism of Intention Revision .......... 16
  2.7 Story Generation Systems .............................. 18
    2.7.1 Mental Modeling .................................. 18
    2.7.2 Personality-Based Behavior ....................... 19
    2.7.3 Goal Extension .................................. 20
    2.7.4 Suspense Creation ................................ 21

Chapter 3 IRIS Generation Process ....................... 23
  3.1 Introduction .......................................... 23
  3.2 Definitions .......................................... 24
  3.3 Suspense Templates ................................. 28
  3.4 Drama Manager Actions ................................ 28
  3.5 IRIS Generation Algorithm ............................ 28
    3.5.1 Overview ......................................... 28
    3.5.2 IRIS’s Belief/Desire/Intention Framework (Initialization) ........ 29
    3.5.3 Protagonist Plan Creation (Step 1) ................ 30
    3.5.4 Application of Suspense Templates (Step 2) ........ 30
    3.5.5 Drama Manager Action Addition (Step 3) .......... 33
    3.5.6 Belief Update (Step 4) ............................ 34
    3.5.7 Intention Revision (Step 5) ....................... 35
    3.5.8 Repair of Suspense Templates (Step 6) .......... 36

iv
LIST OF TABLES

Table 4.1  p values for IRIS suspense templates versus the five other creation processes in the initial experiment. A one-tailed Chi-Squared test was performed versus the random baseline, and a two-tailed Chi-Squared test was performed versus the others. A dash indicates that more than 1/3 of the responses rated neither as suspenseful. ........................................ 48

Table 4.2  p values for IRIS suspense templates versus the three other creation processes in the follow-up experiment. A one-tailed Chi-Squared test was performed versus the random baseline, and a two-tailed Chi-Squared test was performed versus the other two. A dash indicates that more than 1/3 of the responses rated neither as suspenseful. ........................................ 53

Table 5.1  The results of the Mann-Whitney U tests comparing different versions of the game for each of the three suspenseful goals. The one-tailed Mann-Whitney U test was used in the comparisons involving Version 4 and the two-tailed test for comparisons not involving Version 4. The values in bold indicate significant difference with regards to reported suspense for that survey question, p = 0.05, and the values in parentheses indicate marginal significance, p = 0.10. ........................................ 60

Table 5.2  p values for the two-tailed Chi-Squared test comparison of Version 4 and Version 5 of the interactive text game. The numbers in parentheses indicate p values resulting from Barnard’s exact test where Chi-Squared could not be applied. There is not a significant difference between the two versions in terms of player suspense, but there is improvement in Version 4 over Version 5 in the dynamite goal, Question 2, and Question 5. ............... 64

Table 5.3  The results of the one-tailed Mann-Whitney U tests comparing Version 5 to Versions 1-3 of the game for each of the three suspenseful goals. The values in bold indicate significant difference with regards to reported suspense for that survey question, p = 0.05, and the values in parentheses indicate marginal significance, p = 0.10. ........................................ 65
LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure Reference</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 2.1</td>
<td>The syntax for Cohen and Levesque’s formalism of intentionality</td>
<td>14</td>
</tr>
<tr>
<td>Figure 3.1</td>
<td>The IRIS story generation system. The generation process creates story outlines containing suspense that can then be instantiated in different media.</td>
<td>24</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>Algorithm 1: IRIS-Narrative-Generation Algorithm</td>
<td>29</td>
</tr>
<tr>
<td>Figure 3.3</td>
<td>List of the validated IRIS suspense templates. These templates consist of a series of Drama Manager actions, that when inserted into the protagonist’s plan before a given protagonist goal, create dramatic irony and suspense around the goal. For more information on the Drama Manager actions, see Section 3.5.5.</td>
<td>32</td>
</tr>
<tr>
<td>Figure 3.4</td>
<td>Algorithm 2: Intention-Revision Algorithm</td>
<td>35</td>
</tr>
<tr>
<td>Figure 3.5</td>
<td>The current protagonist plan before the suspenseful action sequence is inserted.</td>
<td>37</td>
</tr>
<tr>
<td>Figure 3.6</td>
<td>The preconditions for the suspenseful actions being threatened. An antagonistic character will take a series of actions that threatens the completion of one of these preconditions.</td>
<td>38</td>
</tr>
<tr>
<td>Figure 3.7</td>
<td>Antagonistic plans that can be generated to invalidate the different preconditions of the suspenseful action.</td>
<td>39</td>
</tr>
<tr>
<td>Figure 3.8</td>
<td>Different plans the protagonist can adopt to repair the violated preconditions of the suspenseful actions.</td>
<td>40</td>
</tr>
<tr>
<td>Figure 3.9</td>
<td>The current protagonist plan after the suspenseful actions sequence is inserted.</td>
<td>41</td>
</tr>
<tr>
<td>Figure 3.10</td>
<td>The protagonist plan after the character performs intention revision due to a failed action.</td>
<td>42</td>
</tr>
<tr>
<td>Figure 3.11</td>
<td>The complete set of actions for the suspenseful goal.</td>
<td>43</td>
</tr>
<tr>
<td>Figure 4.1</td>
<td>A visualization of the comparison between the IRIS suspense templates and the three other creation processes in the initial experiment. Red indicates the the column was rated higher and black that they were not significantly different, both at p = 0.05 level. Gray indicates that more than 1/3 of the responses rated neither as suspenseful.</td>
<td>47</td>
</tr>
<tr>
<td>Figure 4.2</td>
<td>A visualization of the comparison between the IRIS suspense templates and the three other creation processes in the follow up experiment. Red indicates the the column was rated higher, green that the row was rated higher, black that they were not significantly different, all at p = 0.05 level. Gray indicates that more than 1/3 of the responses rated neither as suspenseful. One comparison was right at this 1/3 but had a highly significant p value.</td>
<td>52</td>
</tr>
<tr>
<td>Figure 5.1</td>
<td>A visual representation of the western heist game world.</td>
<td>58</td>
</tr>
<tr>
<td>Figure 5.2</td>
<td>A screenshot of the GUI that the participants used to play the game.</td>
<td>59</td>
</tr>
</tbody>
</table>
Figure 5.3  A visualization of the results of the one-tailed Mann-Whitney U tests comparing Key and Loot goals in Version 4 to those goals in the other three versions. A green box indicates that Version 4 was rated significantly higher with regards to suspense for that survey survey question, $p = 0.05$, yellow indicates marginal significance, $p = 0.10$, and black indicates no significant difference. ................................. 62

Figure A.1  List of the initial IRIS suspense templates. The templates with an asterisk were not rated as suspenseful in the evaluation. ................................. 78

Figure A.2  The sentence bank for phase 1 of the initial experiment of IRIS’s non-interactive story generation ................................. 79

Figure A.3  The sentence bank for phase 2 of the initial experiment of IRIS’s non-interactive story generation ................................. 80

Figure A.4  The sentence bank for phase 3 of the initial experiment of IRIS’s non-interactive story generation ................................. 80

Figure A.5  The template-based grammar for story fragment creation in the follow-up experiment of IRIS’s non-interactive story generation ................................. 81

Figure A.6  The background information given to the participants about the story world and the desire to create suspense. ................................. 82

Figure A.7  A screenshot of the survey, where participants read and compared the suspensefulness of two story fragments. ................................. 83

Figure B.1  The survey questions presented to the participants in the post game survey. These questions were generated by an operationalization of IRIS’s definition of suspense. ................................. 84

Figure B.2  A screenshot showing an example question from the post game survey presented to the participants. ................................. 85

Figure C.1  Part 1 of the actions available in the Western domain, used in the example in Section 3.6 and the evaluation of IRIS in an interactive domain. . . . . 87

Figure C.2  Part 2 of the actions available in the Western domain, used in the example in Section 3.6 and the evaluation of IRIS in an interactive domain. . . . . 88

Figure C.3  Part 3 of the actions available in the Western domain, used in the example in Section 3.6 and the evaluation of IRIS in an interactive domain. . . . . 89

Figure C.4  Example initial state of the world for the Western heist domain. This was the +hints/+suspense version of the game. ................................. 90
Chapter 1

Introduction

1.1 Introduction

In the movie, *The Matrix*, the protagonist Neo is one of a group of rebels trying to bring down the evil mind-control system called the Matrix. At one point in the story, the audience is shown a scene where a rebel named Cypher sells out the others to the agents of the Matrix. This betrayal will result in the other rebels being trapped and captured by the agents. Without knowing the danger, the rebels enter the Matrix on a mission. The audience knows that the rebels’ plan is doomed and feels anxiety due to the upcoming trouble. This anxiety comes to a climax when the plan finally fails and the characters realize the betrayal. The narrative construct that this scene uses is called dramatic irony and is created to cause a feeling of suspense in the audience. In my research, I show that it is possible to computationally create stories with suspenseful scenes like the example described above and I propose that such a system is useful for authors to aid in story creation.

Computational story generation systems are valuable for their ability to procedurally and automatically generate stories that contain interesting narrative phenomena. My thesis research focuses on the creation of a narrative generation system called Intention Revision in Storytelling, or IRIS, which has a particular emphasis in creating stories with suspense. Effective suspenseful story generation system need to have a well founded definition of suspense that is operationalized into the generation process. My IRIS system draws its definition of suspense from a successful existing generation system called Suspenser [8]. In addition, the generation system should be effective at creating stories for a variety of narrative media. The IRIS system has been validated at creating suspense in non-interactive text stories and interactive text adventure games. IRIS is a useful tool for narrative authors who can take the suspenseful story output from the system,
1.2 Motivation

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.” [28] 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, [32] as well as they are usually constructed to conform to some commonly accepted dramatic structure. This is opposed to non-story narratives, such as news articles and police reports, which mainly consist of the communication of facts to the audience.

To this day, stories are still almost exclusively hand-authored. Systems that automatically or procedurally generate stories are in their infancy, largely due to the complications that creating interesting stories present. At best, the required structure of an interesting story is complex, and at worst, the structure is still unknown. Narratologists like Propp [29], who analyzed the components of Russian folk tales, 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.

One way to model these believable characters is to represent their mental model using the cognitive belief/desire/intention (BDI) framework [2]. 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 representing speech acts [9], but when BDI is extended to include the concept of intention revision, it can be applied to narrative generation systems. I describe here one such system— the 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 or intentions due to a change in its beliefs. IRIS is a narrative planner that provides rules for adopting and dropping intentions in the context of narrative and uses a BDI mental model for the protagonist of the story.

One powerful way to leverage intention revision in a narrative generation system is to have the intention revision aid in the creation of suspense. Suspense is “the feeling of excitement or anxiety that audience members feel when they are waiting for something to happen and are uncertain about a significant outcome” [8]. Since IRIS’s generation process provides a framework for instigating or preventing intention revision, IRIS has direct control over the manipulation of the individual characters’ plans and goals, which it uses in conjunction with psychological and narrative models of suspense to create high suspense stories.

To create this feeling of suspense in the audience, the IRIS system focuses on creating suspense around the protagonist of the story. IRIS’s model of the protagonist has a robust BDI mental model that controls its planning and intention revision. After the protagonist forms a plan, the generation process introduces threats to certain goals of the protagonist’s plan using a set of suspense action templates. These suspense action templates were created for the purpose of the system and were experimentally validated. The suspense action templates create suspense at the story-action level, as opposed to the discourse level, by eliminating possible paths of success to achieving the target goal. The other, non-protagonist characters, called supporting characters, are available to help perform these actions that are interjected by the generation process. In the case of plan failure due to these threats, the protagonist performs intention revision to modify its plan and intentions. The process repeats until a complete suspenseful story is generated.

The intention of the IRIS system is to produce a story template that has suspense intrinsic to it at the story-action level and also has believable protagonist behavior. The output of the
IRIS system can then be applied to both non-interactive and interactive contexts. For my thesis, story fragments were chosen as a non-interactive medium and a text adventure video game was chosen as the interactive medium. I show that the story outlines produced by the IRIS system create suspense in the audience in both these two different contexts.

1.3 IRIS Story Generation Process

The IRIS system uses a process called planning to generate a suspenseful story outline. Planning is an Artificial Intelligence technique that guides that selection and ordering of actions into a coherent, ordered action sequence. The planning process is centered around a central character called a protagonist. The system captures the protagonist’s mental state, which contains information like the protagonist’s beliefs and goals. As the character works to complete personal goals and authorial imposed goals, the system interjects suspenseful action sequences meant to complicate the completion of certain story actions. These action sequences create suspense by altering the audience to the protagonist’s imminent plan failure which is unknown to the audience. The audience also experiences suspense as the protagonist struggles to find other ways of completing its plan. The output of the IRIS system is a suspenseful story outline that contains these moments of suspense. An author can take this output, add discourse details to the outline, and have a complete, suspenseful, story.

1.4 Contributions of the IRIS System

The contributions of the IRIS system are twofold. First, the system is a narratively driven planner that creates suspenseful story outlines. Second, the outlines it creates can be translated to a variety of different media, like non-interactive text and interactive text games.

One of the greatest benefits of computer science is its interdisciplinary ability to formalize, computationalize, and automate human processes. In the field of narratology, there is a drive to study and formalize the fundamental narrative phenomena. One such phenomenon of interest to my work is suspense. The IRIS system uses one particular definition of suspense: the feeling of excitement or anxiety that audience members feel when they are waiting for something to happen and are uncertain about a significant outcome [8]. This definition is applied to one subset of suspense that can be created in a story: story-action level suspense created from protagonist plan failure. The IRIS system computationalizes this definition and uses planning to generate a story template whose action selection and ordering creates this kind of suspense in a reader.
These story templates can also be translated into different media while preserving the sense of suspense from the initial outline. The ability for the outlines to be used in different media allows for a novel author to use the system to create his or her story outline, which can then be modified to allow for the author’s personalization. The flexibility of the output also allows for the automation of suspense creation in a game world. The outline created by IRIS system provides an alternative for a game designer to writing all of the suspenseful story moments ahead of time into the game. Additionally, the template nature of the output of the IRIS output allows for flexibility in its instantiation into the game world, allowing choice of how the suspense is realized in the game.
Chapter 2

Related Work

2.1 Introduction

The IRIS system computationally generates suspenseful story outlines. The suspense is created around a central protagonist character and suspenseful event sequences are interjected into the protagonist’s plan to complicate the completion of chosen story goals. The audience is aware that these malicious actions will cause the character’s plan, in its current state, to fail, but the protagonist is unaware of the upcoming failure. This imbalance of knowledge causes a feeling of suspense in the audience as the character unknowingly carries on its plan and finally comes to realize that it needs to modify its plan to have its goal succeed.

To generate these suspenseful story outline, the IRIS system draws upon a variety of related work. It makes use of narrative theory and narrative’s way of communicating fundamental emotions and data to the readers. Specifically, suspense is created in the user through dramatic irony, which refers to the imbalance of information between the audience and the protagonist. The system creates suspense this way when it shows the audience the protagonist’s upcoming plan failure without the protagonist knowing the complication.

To actually create the sequence of story actions, IRIS uses the Artificial Intelligence technique of planning. Planning refers to the selection and ordering of actions (in this case story actions) that achieve a set of goals. The goals in this case consist of both protagonist goals and authorial imposed goals. The protagonist’s plan may change over the course of the story as it gains new information and changes how it wants to achieve its goals. However, final product of the generation process is a complete story plan where all of the important goals are achieved.

To guide the planning process, the protagonist is given a cognitive mental model called the Belief/Desire/Intention framework. This helps guide the agent balance its wants and actions.
The segmentation of knowledge between the character and the true state of the world is important to achieve dramatic irony and create suspense in the audience. The protagonist will sometimes embark on plans that the audience knows will not succeed, and this is because the protagonist does not know the true state of the world at the time.

IRIS is not the first system to computationally generate stories. IRIS draws inspiration from these related systems that use a variety of techniques in story creation like modeling characters’ mental models, selecting actions to match character behavior, and using authorial goals to generate more goals for the system to satisfy. Some systems even use planning or generate suspense. However, IRIS provides a novel contribution because it is the first narrative planning generation system that creates suspense in the action selection and ordering.

2.2 Understanding Narrative

Narratives are everywhere in the world: movies, books, TV shows, and even commercials contain narrative. It is easy to see why it could be useful to produce a system that alleviates some of the authorial burden of narrative creation. Fundamentally, narrative is the perceptual activity that organizes data into a special pattern which represents and explains experience [1]. It is an important linguistic phenomenon that helps people make sense of the world [25]. Narrative is the common tool used by humans instinctually and unconsciously to sequence related events and actions into their memory. Though narratives can be constructed in many different ways, the most common form of narrative is the transformation of one thing to another and the interesting attributes that can describe this transformation [1].

Narrative is also an effective way of understanding data and provides a significant boost to comprehension. Mateas and Sengers state that narrative is built in to human cognition and that people are narrative animals [25]. Studies have confirmed narrative’s ability to convey information. For example, narratives can help people process unsorted data [1]. Also, narrative dramatically decreases reading times of text and dramatically increases recall [17]. Thus, narrative has some very attractive features that can be leveraged for increased information comprehension.

Though commonly used interchangeably, narrative and story refer to separate concepts. Narrative consists of two components: fabula and syuzhet. These terms were first used by Propp when analyzing Russian folk tales [29]. Fabula, also called story, is a temporally ordered sequence of events created by the author. It is independent of style, language, and medium [32]. Syuzhet, also called narration or discourse, represents a telling of some fabula. It can change from storyteller to storyteller, or even among tellings from the same storyteller. An
infinite number of syuzhets can be generated from the same fabula [32]. In fact, the way the syuzhet is presented can change the genre of the story. A suspenseful story is one where the storyteller gives the audience more knowledge than the character. A mystery story is one where the knowledge is equal, and a surprise story is one where the audience knows less than the characters [1]. Thus the choice of syuzhet can dramatically change the audience’s experience.

Though narratives can be simple or complex, most narrative fiction follows a simple structure. The narrative consists of a series of temporally ordered events, each of which opens a new possibility in the story or prolongs a current one [32]. Let $A$ and $B$ represent conditions of the world over the course of a narrative. The typical fictional narrative can be represented in these five stages: $A$, $B$, $\neg A$, $\neg B$, $A$ [1]. For example:

- A boy had a ball. ($A$)
- A dog came and took the ball. ($B$)
- The boy no longer had the ball. ($\neg A$)
- The boy’s mom took the ball away from the dog. ($\neg B$)
- She gave it back to the boy. The end. ($A$)

Since narrative is such a broad phenomenon, I choose to focus on creating narratives with a specific property: suspense. Though well researched in narrative and psychological literature, suspense has not been heavily studied in narrative generation. The IRIS system intends to bring suspense more to the forefront in narrative generation, since it produces a very unique and desirable feeling in the audience.

2.3 Suspense in Narrative

Whether watching Alfred Hitchcock or horror, it is suspense that drives the movie. Yet for such a popular narrative phenomenon, categorizing suspense is more difficult than one would expect. There are many different definitions of suspense and not one unified, agreed upon definition. For example, Rodell, a novelist, describes suspense as “the art of making the reader care about what happens next” [34]. The definition the IRIS system uses comes from an existing narrative generation system called Suspenser: “the feeling of excitement or anxiety that audience members feel when they are waiting for something to happen and are uncertain about a significant outcome [8].” This definition was chosen because it has already been successfully computationalized in a planning system.
Suspense is the careful interplay between the author and the audience, where the author strategically reveals information to the audience to heighten anxiety building up to an important event which has an uncertain outcome. Interestingly, this anxiety has a negative valence, yet it is clear that people seek out and enjoy suspense in narrative [23]. There are several theories that attempt to address this paradox, such as arousal jag, where experiencing an intense emotion has itself positive valence, and arousal boost, where the payoff of a resolved experience of suspense is worth the temporary negative emotion. Nevertheless, suspense in some form is present in almost every well-constructed narrative.

It is important to distinguish suspense from other major narrative phenomenon, such as mystery and surprise. An important distinction concerns the discourse of story events. In a narrative with six story events (labeled A1 to A6), the discourse of each of suspense, mystery, and surprise might look like:

Suspense: A1 A2 A3 A4 A5 A6
Mystery: A2 A3 A4 A5 A6 A1
Surprise: A2 A3 A1 A4 A5 A6

Thus, in a suspenseful story, the discourse and story structure can be the same, as long as there is an initiating event for the suspense (in this example, A1) [3]. In a mystery, the audience is looking backward to learn about an unknown earlier event, where in suspense the audience is looking forward to an upcoming unresolved outcome. Additionally, in a mystery, there may be many outcomes to this past event. However, in a suspense, there are only two: success or failure or the suspenseful goal [5]. Finally, a surprise has small past uncertainties that are resolved over the course of the story.

To successfully create suspense around an outcome, there need to be several conditions in place. First, events are more suspenseful when they involve the protagonist [49]. This is the reason the IRIS system only creates suspense around the protagonist of the story. Second, the viewers need to be invested in the outcome. Third, there has to seem like there is a low probability of success. Finally, the outcome has to be something bad, evil, or morally unsettling [5].

There is no one right way to create suspense, but multiple formula have been proposed. Many of them revolve around demonstrating that there are only a few paths to success for a goal, and then systematically cutting off those paths [8][16]. This is the approach that the IRIS system uses, because of its natural fit with IRIS’s story generation technique called planning. Another approach to creating suspense concerns having the audience identify and empathize with the protagonist [44][49]. Suspense can also be created at the discourse, or narration, level, by showing less detail and allowing the audiences imagination to fill in the detail [47] or by
having the audience anticipate failure, even if possibility of failure is usually low [47].

2.4 Evolution of Narrative Generation

Since narrative and its components, such as suspense, can provide an important effect on the audience, computer scientists have been working over the last few decades to computationally generate narratives. Work in the 1970’s and 1980’s started with the study of knowledge structures and natural language [4, 11, 13, 21, 26, 46]. This work developed into some of the first computational narrative generators. These systems were for the most part heavily knowledge based, meaning that they only functioned in very limited domains. Recently, the study of narrative and narrative generation has reached a broader interdisciplinary context, with tie-ins to art, psychology, culture, literary studies, and drama [25]. Games, too, have received attention for their narrative properties [19]. One useful and popular technique in current work in narrative generation is the adaptation of Artificial Intelligence’s concept of planning.

2.5 Planning

The IRIS system uses planning as it method of selecting and ordering story actions into a complete action sequence. In the final story plan, the protagonist will have completed all of the authorial imposed goals, and there will be moments of suspense created around the completion of these goals. The protagonist has only partial knowledge of the world, so it may try to work towards goals that it cannot achieve with its current plan. It is in the imbalance of knowledge between the protagonist and the audience that the IRIS system creates suspense.

2.5.1 Motivation

Planning is useful for story generation because of the natural mapping between actions in a plan domain and story events. It is a systematic and repeatable process to transform the initial state of a story to the desired end state. We can also easily adapt existing planners for the narrative generation problem.

2.5.2 Description

Planning is the task of coming up with a sequence of actions to achieve a goal [36]. A planner is given an initial description of the world, a description of the desired end state of the world, and a set of actions. The output of the planner is a sequence of actions that will achieve the
goal. There are a number of assumptions that are commonly made when performing planning. They are atomic time of the actions, deterministic effects, omniscience of world space, and that the planner is the sole cause of change [45].

Planning can be cast as a search problem through plan spaces. Each plan space node in the search space represents a set of partially instantiated actions and a set of constraints on those actions. Each action is represented by a set of preconditions and effects, where the preconditions are the conditions that must hold to perform the action and the effects are the change that the action brings about. The planner keeps track of the actions and their causes and effects.

Constraints on the actions are of two types. One kind of constraint is an ordering constraint, where one action must precede another. The other kind of constraint is a causal constraint. A causal constraint is represented in the form $A \rightarrow^C B$, meaning that the precondition $C$ of action $B$ is satisfied by action $A$.

Any inconsistency that the current plan has with the final state is called a flaw. The flaws are corrected one at a time in a process called refinement [20]. There are two types of flaws: threatened links and open preconditions. In the case that an action comes between actions $A$ and $B$ and undoes a precondition of $B$, the link is said to be threatened. The threatened causal link can be resolved by either putting the threatening action before $A$ (demoting) or after $B$ (promoting) [45]. An open precondition exists when one of the preconditions of the final state is not satisfied. Once all flaws have been corrected, the plan can be given as the series of transformations that shape the initial state into the final state.

Modifications on this standard planning algorithm can be introduced to add flexibility. Some algorithms allow for action schemata with variables, conditional effects, disjunctive preconditions, and universal quantification [45]. Planners can also be modified to operate in dynamic, changing environments.

In the case of dynamic environments, it is often helpful to postpone parts of the planning process and complete these parts of the plan when more information has been gathered about the environment. Four different methods of dealing with a dynamic environment are: 1) sensorless planning, which forces the world to cooperate so that the planner can achieve its goal in all possible circumstances, 2) conditional planning, which senses the world, constructs different branches of the plan for different contingencies, and selects one, 3) execution monitoring and replanning, where a plan can be monitored and revised, and finally 4) continuous planning, which is a true mix of planning and execution [36]. The intention with all of these techniques is to allow a single planner to delay some of the planning in a dynamic world to prevent constant backtracking due to the world changing. However, it is also important when an agent is planning to negotiate with other agents that can have an effect on the agent’s plan. This is
2.6. BELIEFS, DESIRES, AND INTENTIONS

There are several different levels of collaborative planning. The first is a master-slave assumption, where one agent directly controls another through these utterances. This method is limited because it does not allow for robust collaboration [18]. In the second kind of collaboration, an agent tries to provide enough information to other agents in order to convince the other agents to adopt its preferences [12]. Finally, the kind of collaboration that allows for the most freedom among agents allows for multiple agents to have a shared plan. All of the agents need to have the same mutual beliefs, and can perform joint actions as long the action is in each agent’s plan [18]. No matter what collaborative planning method is chosen, it is important in multiagent planning systems to consider how the agents will interact and share information.

The expressive power of utterances [18] can be harnessed in collaborative planning to help convey information from one agent to another in order to influence that agent to adopt a plan. One kind of utterance is a statement that a speaker communicates to a listener in order to provide information to the listener or request information for the speaker. One utterance does not always correspond on a one-to-one ratio with one desired action. A single utterance might imply a commitment to multiple actions [24].

2.6 Beliefs, Desires, and Intentions

The belief/ desire/ intention (BDI) framework allows for a segmentation between the true and complete state of the world and the protagonist’s mental knowledge. It is a way of capturing the protagonist’s motivations, desires, and wants. The protagonist may have erroneous beliefs about what is true in the world; in fact this is an important aspect of the IRIS story generation process. A malicious character will take actions to thwart the protagonist’s plan. These actions will be known to the audience and unknown to the protagonist. The knowledge that the protagonist is doomed in its current plan creates a feeling of suspense in the audience. However, the protagonist will carry onward with its ill-fated plan due to the power of the segmentation of knowledge that the BDI framework affords.

2.6.1 Motivation

BDI cognitive modeling is used in IRIS because of the desire to give identifiable motivation to the protagonist’s actions. The BDI framework also is a good way to model the protagonist’s changing goals throughout the course of the story. The character goals can be integrated with the planning system’s end state and story actions will be chosen by the planner that show the
protagonist’s progress towards its goals.

2.6.2 Description

One way to model a computer agent’s cognitive model is with a BDI framework. This cognitive model allows for a clear distinction among an agent’s knowledge, wants, and goals. Cohen and Levesque were among the first researchers to logically formalize the BDI framework [9]. Since then, there have been many extensions to their formalization. The different extensions to the core BDI formalism afford different research applications. Singh [38], for example, made changes to Cohen and Levesque’s formalization of intentions. This extension addressed problems with adopting and dropping intentions and allowed for multiple intentions. Another extension is multiagent communication [9]. Finally, van der Hoek’s [42] extended the definition of beliefs and introduced a logical formalism of intention revision.

The three core components of the BDI framework are beliefs, desires, and intentions. 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. 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 [42]. For example, an agent could have the desire to become a millionaire, but not believe that this desire is possible. 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 [10].

2.6.3 Cohen and Levesque’s Formalism of Intentionality

To explain their logical formalism of intentionality, Cohen and Levesque first define operators to represent the BDI framework and any other important information they need to capture their representation. Figure 2.1 shows the syntax that they use. Especially of interest are BEL, GOAL, and the “time” proposition. They define operators GOAL and BELieve to capture the beliefs and goals of an agent. Since desires can be inconsistent, they do not define a desire operator [10]. However, Cohen and Levesque’s use of goals and beliefs bear some discussion.

There are several kinds of goals an agent can have. The first is called an achievement goal.
\(< \text{Action} - \text{var} > ::= a, b, a_1, a_2, ..., b_1, b_2, ..., e, e_1, e_2, ... \)

\(< \text{Agent} - \text{var} > ::= x, y, x_1, x_2, ..., y_1, y_2, ... \)

\(< \text{Regular} - \text{var} > ::= i, j, i_1, i_2, ..., j_1, j_2, ... \)

\(< \text{Variable} > ::= \langle \text{Agent} - \text{var} \rangle | \langle \text{Action} - \text{var} \rangle | \langle \text{Regular} - \text{var} \rangle \)

\(< \text{Pred} > ::= \langle \text{Pred} - \text{symbol} \rangle | \langle \text{Variable} >_1, ..., \langle \text{Variable} >_n \rangle \)

\(< \text{Wff} > ::= \langle \text{Pred} \rangle | \neg \langle \text{Wff} \rangle | \langle \text{Wff} \rangle \lor \langle \text{Wff} \rangle | \exists \langle \text{Variable} \rangle \langle \text{Wff} \rangle | \langle \text{Wff} \rangle \)

one of the following:

\(< \text{Variable} > ::= \langle \text{Variable} \rangle \)

(HAPPENS \(< \text{Action} - \text{expression} \>): \(< \text{Action} - \text{expression} \>) \text{ happens next},

(DONE \(< \text{Action} - \text{expression} \>) \text{ has } \text{just} \ \text{happened},

(AGT \(< \text{Agent} - \text{var} >\langle \text{Action} - \text{var} \rangle \): \(< \text{Agent} - \text{var} \>) \text{ is the } \text{only} \ \text{agent of}

\(< \text{Action} - \text{var} >,\)

(BEL \(< \text{Agent} - \text{var} >\langle \text{Wff} \rangle \): \(< \text{Wff} > \) \text{ follows from } \(< \text{Agent} - \text{var} >\)'s beliefs,

(GOAL \(< \text{Agent} - \text{var} >\langle \text{Wff} \rangle \): \(< \text{Wff} > \) \text{ follows from } \(< \text{Agent} - \text{var} >\)'s goals,

\(< \text{Time} - \text{proposition} > ::= \langle \text{Numeral} \rangle \)

\(< \text{Action} - \text{expression} > ::= \langle \text{Action} - \text{var} \rangle \) \text{ one of the following:}

\(< \text{Action} - \text{expression} >; \langle \text{Action} - \text{expression} \rangle \): \text{ sequential action},

\(< \text{Action} - \text{expression} > - \langle \text{Action} - \text{expression} \rangle \): \text{ nondeterministic choice action,}

\(< \text{Wff} > ?\): \text{ test action,}

\(< \text{Action} - \text{expression} >^* \): \text{ iterative action}

**Figure 2.1** The syntax for Cohen and Levesque’s formalism of intentionality
An achievement goal is one that the agent currently believes is false and intends to become true. Once an achievement goal is satisfied, it is dropped. The second type of goal is a maintenance goal. Maintenance goals are those that the agent currently believes to be true, and intends to perform the necessary actions to keep them true. Maintenance goals are not usually dropped. The third type of goal is called a persistence goal (or P-GOAL). Persistence goals are established to make sure the agents do not give up their goals too easily. They are not dropped until the agent believes the goal to be satisfied, or until it thinks the goal will never be true.

There are several desirable behaviors of BDI-based agents with respect to goals. In general, it is not desirable for an agent to either hold a goal forever, or to put off accomplishing the goal forever. Therefore, most goals will have limited persistence and will eventually be dropped. That being said, the agent cannot deliberately act to make one of its goals unachievable. The agent will ideally formulate a goal, eventually achieve the goal, and then drop the goal.

Cohen and Levesque similarly adopted the previously explained concept of beliefs with some modifications. First, as explained before, beliefs do not have to be consistent with the actual state of the world. Second, the agent’s beliefs will constrain its intentions, since the agent cannot choose plans which are incompatible with its beliefs. This implies that if an agent believes p is true now, it cannot intend for it to be currently false, since the agent cannot choose what it cannot change. Finally, a belief can prevent the agent from taking an action entirely. If agent A believes that another agent B will definitely achieve A’s goal, A will not form an intention to achieve the goal.

There are several properties of intentions as defined by Cohen and Levesque. They state that “intentions normally pose problems for the agent; the agent needs to determine a way to achieve them,” and that if an intention does not somehow influence the agent’s actions, then there is a problem with the modeling of either the agent or the intention. Additionally, there are two kinds of intentions, and both affect the agent in certain ways. “Present-directed” intentions dictate the agent’s current behavior and actions, as explained before. However, “future-directed” intentions guide the agent’s planning and constrain the adoption of other intentions. For example, if the agent has the intention to leave the country for a week, then it will not also form the intention to attend the next-door neighbor’s house party that week. Finally, if an agent fails to achieve its intentions, it should not try the same plan over and over if there is no chance of success. There should be some mechanism for replanning and the generation of a new plan that the agent believes will succeed. Cohen and Levesque do not propose how such replanning can be done, but previous work by van der Hoek, described in Section 2.6.4 and my work, described in Chapter 3, both suggest a methodology for replanning. Intentions are also closely related to the agent’s beliefs and goals. For an agent to intend to achieve a goal p, four conditions must
hold. First, the agent has to believe that $p$ is possible, that is, it believes that in its world $p$ could become true. Second, the agent does not believe that he will not bring about $p$, meaning it cannot intend $\neg p$. Third, it believes there is some series of actions that the agent can perform that can bring about $p$. Finally, the agent does not have to intend $p$’s side effects. Thus, the agent can intend to make a cake without the intention of using up all of the butter, becoming fat, or messing up the kitchen.

### 2.6.4 Logical Formalism of Intention Revision

Van der Hoek [42] expands Cohen and Levesque’s notion of intentionality to include intention revision. Intention revision is important for my research because the way the IRIS system creates suspense is through plan failure. It is possible that the protagonist will need to modify its plans or intentions after the system introduces plan failure, so its BDI mental model should include the ability to perform intention revision. Van der Hoek’s work is in response to Cohen and Levesque’s theory discussing the components of intentionality, but not its dynamics, specifically, how an agent should rationally adopt and drop its intentions. Van der Hoek states that although intentions are a key component of rational agents, intention revision has hardly been considered. However, since intentions are closely related to other mental states, such as beliefs and desires, they cannot be studied in isolation. Therefore, van der Hoek makes a few extensions to the standard BDI framework.

Desires are described the same way as in Cohen and Levesque: states of the world that the agent would ideally like to have become true. Intentions are commitments to action that give rise to goals. Van der Hoek explicitly defines his version of persistent goals. An agent has a persistent goal $p$ if it has a goal that $p$ will eventually become true but is not true right now, and before it drops this goal either the agent believes that the goal has been satisfied or it believes that the goal will never be satisfied.

His version of beliefs are also expanded from the Cohen and Levesque formalism. Like before, they represent what the agent thinks to be true. However, the notion of weak and strong beliefs is introduced. Weak beliefs, or WB, “represent what the agent should believe under the working assumption that he will succeed” [42]. Thus, if the agent intends to plan to play basketball, it works under the assumption that it will at some point have succeeded in playing basketball. It does not assume that it will fail. Strong beliefs, sometimes called SB or pessimistic beliefs, only refer to the belief operators in the agent’s set of beliefs. Under strong beliefs, the agent does not make any special assumptions about what it might or might not be able to accomplish. A change to the strong beliefs can cause a change in intentions, which would then cause a
change in weak beliefs. For example, consider again an agent intending to play basketball in
the afternoon. It has some set of strong beliefs about the world, and the weak belief that it
will succeed in playing basketball later. If the agent then finds out it will rain in the afternoon,
it can no longer intend to play basketball. Therefore, the weak belief that it will succeed in
playing basketball in the afternoon also changes.

Van der Hoek defines intentions as the desires that the agent is committed to bring about. Van
der Hoek lists the following as additional desirable properties of intentions: agents should
actually act on their intentions, not in spite of them; the agents should adopt feasible intentions
but avoid infeasible ones; agents should keep their intentions, but not forever; agents should
drop the intentions once they have been satisfied; agents should alter their intentions as their
beliefs change (with regards to strong beliefs, as described above), and finally agents should
adopt subsidiary intentions during plan formation. Though the details of van der Hoek’s actual
formalism are beyond the scope of this paper, he describes a language that achieves the above
requirements. The language attempts to include the ability for the agent to revise its intentions.
There certainly seem to be instances where such intention revision is necessary, such as when
the agent believes the goal has been fulfilled, or that it can never be fulfilled. However, van
der Hoek believes that it is important to “disturb the mental state as little as possible” [42].
Therefore, intention revision should only come from stimulus from outside of the agent itself,
such as the introduction of new information or the passage of time, and not spontaneously from
the agent itself. These safeguards complete his model of intention revision.

An example using van der Hoek’s formalism will be presented to show how he handles inten-
tion revision. The example agent desires include wanting a nice, continental, English breakfast,
if there is no cereal, then to have yogurt, to be energetic tomorrow, and to be happy throughout
the week. His representation (with o meaning “in the next step” or “tomorrow”) looks like:

\[\text{Desires} = \{\text{o nicebf, o contb, o Englishbf, o energetic, o (\neg \text{cereal} \rightarrow \text{yogurt}),}\]
\[\Lambda_{i=1...7} \text{o}^i \text{ happy}\}\]

Next, his reasoning rules for accomplishing these goals:

\text{P1: nicebf }\leftarrow \text{ milk, cereals, nicecoffee}
\text{P2: nicebf }\leftarrow \text{ milk, cereals, tea}
\text{P3: happy }\leftarrow \text{ newspaper}
\text{P4: happy }\leftarrow \text{ egg}
\text{P5: energetic }\leftarrow \text{ nicecoffee}
\text{P6: }\neg \text{happy }\leftarrow \text{ bill, nicebf}
\text{P7: nicecoffee }\leftarrow \text{ espresso}

Two plans can be expressed as follows:
Plan 1: o nicebf → o milk, o cereals, (o nicecoffee → o expresso)
Plan 2: o energetic → (o nicecoffee → (o espresso))

Van der Hoek also offers an example of intention revision using his formalism. If the agent discovers that it cannot obtain milk, meaning ¬o milk is added to the beliefs, it should no longer plan on o nicebf. But though it drops o milk, it should still try to accomplish o nicecoffee, since that will satisfy Plan 2 of being energetic. This example illustrates several things. First, it provides an example of intention revision in van der Hoek’s formalism. Second, it shows that if a plan can’t be completed, it does not necessarily mean that an agent should abandon the plan entirely. Finally, the example shows how the beliefs and plans are carefully revised and how the agent’s mental model is disturbed as little as possible.

Van der Hoek sought to expand upon Cohen and Levesque’s model of intentionality. He observed that Cohen and Levesque did not develop a method in their language by which an agent could properly adopt or drop intentions. Van der Hoek developed an expansion upon the original Cohen and Levesque framework that added a method for intention revision. This included an expanded notion of intentions and temporal operators that attempted to alter the agent’s mental state as little as possible. With these safeguards in place, van der Hoek has tried to model a robust language that allows for the proper adoption and release of intentions.

2.7 Story Generation Systems

There have been numerous systems that have been designed to computationally generate stories. Each system emphasizes different aspects of the narrative problem. The IRIS system draws inspiration from these systems but also has its own unique focus.

2.7.1 Mental Modeling

A number of the narrative generation systems have focused on the mental modeling of the characters in the story. The researchers focus on having their story characters act in coherent and believable manners and take actions that are consistent with their expected behavior.

2.7.1.1 A Cocktail-Party Simulation

Strong et al. [39] observe that most conversation in games is hand authored, and designed a system that would try to allow for system level support of conversation generation in computer agent/computer agent interaction and computer agent/ player interaction. Their system offers an alternative approach to BDI cognitive modeling. Each agent’s mental state is modeled as
a graph, with the nodes in the graph a topic the agent wants to talk about and the links keywords connecting the nodes. These topics and keywords are loaded in from a prewritten file. In conversation, the agent tries to bring up a topic based on the current topic and previous things that have been said. This approach is effective because it attempts to create a flow of dialogue in the story, however, it is not very procedural, and the conversations produced end up not looking very natural.

2.7.1.2 Social Planning in “Othello”

Chang et al. [7] designed a system to simulate the story of Shakespeare’s Othello. The Iago agent engages in social planning, where the plans require the cooperation of more than one agent. In the system, Iago is the sole manipulator agent that can influence the intentions of other agents, while every other agent is a “patient” and can not exert this influence. The system has two different kinds of action sequences: causal sequences, where each successive action is motivated by the previous action, and problem solving sequences, where all actions collectively will satisfy a goal. In addition to the interesting aspect of modifying other agents’ intentions, Chang’s implementation is valuable because it gives insight into cooperative social plans and the difficulties they introduce.

2.7.1.3 IPOCL

Riedl et al.’s IPOCL [31] produces a story based on the BDI framework. Agents’ goals are noted by the universal story planner and chained backwards in a series of intentions and actions. To fulfill the ending state, the system first motivates a character to want to perform that action, and then the action is added to the story, and so on back to the starting state of the story. The result is a story where the overall plan has agents taking actions that fulfill their intentions to realize their goals. This incorporation of the BDI framework is valuable, but the intentions that are formed to produce the goals are not always coherent. For example, the system could force a character to adopt an intention that was at odds with its behavior throughout the rest of the story. This could result in a story in which the characters have erratic behaviors.

2.7.2 Personality-Based Behavior

Similar to Mental Modeling, the authors of these systems seek to create believable behavior based on a story character’s personality. Characters in these systems will perform actions consistent with their personality features, which can result in different characters achieving the same goal in different ways.
2.7.2.1 I-Storytelling

Cavazza et al. [6] produced a sitcom-generating system where agents’ behavior is generated in real-time with occasional user intervention. The system makes use of an HTN tree, and the planning is interleaved with execution. Also, plans are generated for each agent, instead of a global plan. The planning structure is very useful since it allows for plan failure and replanning. Another feature is that agents behave differently to other agents based on personality, mood, and the agent’s past actions. However, the user interaction is not critical to the system, and the user can only use a very limited set of action to influence the story.

2.7.2.2 UNIVERSE

Lebowitz’s UNIVERSE [22] explores extended stories without definite ends and generates story outlines. Characters have personality traits that guide, or motivate, action selection. UNIVERSE constructs an overarching plan instead of individual agents’ plans, and the planner keeps working until it reaches an outline where the actions are consistent and all of the actions are motivated. This can lead to the generation of multiple plot lines, which increases the expressive power of the system. Another goal of UNIVERSE is to generate plot fragments that can be used across many different stories. The attention to the overarching story plan lends itself well to creating captivating stories.

2.7.2.3 Personality-Based Agents

Rizzo et al. [33] focuses on agents’ personalities. These personalities provide the driving force behind the agents’ decision making. He notes that the same goals can be accomplished by different personalities, but will often have different side effects. Another interesting decision was to have two systems running together: PRODIGY [43] is used offline to compute a library of pre-generated plans, and RAP [15] is used online to adaptively execute plans.

2.7.3 Goal Extension

In these systems, the story is treated as a document that is built iteratively by adding interesting story events. The systems have a model of what constitutes a worthwhile event and the story generation stops once no more interesting events exist.
2.7.3.1 TALE-SPIN

Meehan’s TALE-SPIN [26] is an important early system that recreate fables. The system, given a goal, generates events that achieve that goal and also generates other new goals for the system to satisfy. It also has an assertion mechanism that stores events in memory. TALE-SPIN models different kinds of plans: delta plans, which work towards an overarching goal, sigma plans, which fulfill bodily needs that spontaneously arise, and pi plans, which are plans of preservation and achievement. The system allows for relationships among agents, personalities, and mental mapping of the world.

2.7.3.2 Drama-Generating Plot Manager

Sgourous [37] offers many interesting insights for modeling dramatic story structure and for inter-agent interaction. His Plot Manager iteratively generates possible agent interactions and chooses the most dramatic next step, until there are no more dramatic steps, ending the story. Sgourous identifies at least four patterns of dramatic actions: lifeline, where an unfavorable outcome is followed by a favorable one, rising complication, where there are successive unfavorable outcomes, reversal of fortune, where a favorable outcome is followed by an unfavorable outcome, and irony, where an outcome is favorable for A, who proceeds to bring about an unfavorable outcome for B. These patterns give a way to structure dramatic situations. He also provides a framework for cooperative and competitive social behavior. The agents can intervene to satisfy or impede goals by acting positively or negatively towards other agents. They can also have a positive or negative relationship with the other agents.

2.7.4 Suspense Creation

Some story generation systems that have modeled suspense in the generative process. The MINSTREL system is a case-based reasoning system that creates a number of narrative phenomena including suspense. Suspenser is a planning system that also generates suspense, but unlike the IRIS system it creates suspense at the story discourse level.

2.7.4.1 MINSTREL

MINSTREL [41] is a case-based reasoning system that creates stories that have, among other properties, suspense. It creates suspense mainly at the story-action level, as opposed to the discourse level. MINSTREL has a simplistic model that determines when and how to create this suspense. The only appropriate time to create suspense in its model is in a scene that the
author wants to emphasize, and when the character’s life is in danger. MINSTREL suspense formula is the following: 1) Make the target character sympathetic. 2) Choose an important scene. 3) Threaten that character’s life. 4) Interject emotion at the discourse level. 5) Have the character try to get away and fail. MINSTREL is valuable because it provides a framework for when, when not, and how to create suspense around an outcome. Its weakness is the overly simplistic method of creating suspense, likely because many different narrative phenomena were present in the system and suspense was not the focus of the work.

2.7.4.2 Suspenser

Suspenser [8] is a planning system that generates suspenseful stories. It generates suspense at the discourse level. Suspenser first uses a planner to produce the story events. It then iteratively increases suspense by replacing and reordering story events to create a discourse. Suspenser is the closest system to IRIS in that they are both planning systems that focus on creating suspenseful stories. However, since Suspenser generates suspense at the discourse level, it is not generating stories that are intrinsically suspenseful, but rather adding suspense to the stories after they have been generated. Suspenser also is not guaranteed to generate high suspense stories, since it can get stuck in a local maximum when adding suspense. Finally, the goal of Suspenser is to generate stories with as much suspense as possible, but it is not clear that this is desirable.
Chapter 3

IRIS Generation Process

3.1 Introduction

The Intention Revision in Storytelling (IRIS) system is a story generation system designed to create suspenseful story outlines. An overview of the system is shown in Figure 3.1. The outer orange box represents the system as a whole. The user of the system provides IRIS the initial story domain information, such as actions that can happen in the story and a list of characters.

This information is used by the system’s planner, which is represented by the blue box. A planner is a tool that allows for action selection and sequencing, and when applied to story generation, story action selection and ordering. The IRIS planner creates the story plan around the protagonist, who is at the focal point of the story. To allow for the protagonist to behave in a believable and coherent manner, the protagonist is provided with two psychological properties. The first is a Belief/Desire/Intention (BDI) framework, which allows for the protagonist to have its knowledge and beliefs segmented from the rest of the world. The second is a model of intention revision, which dictates how the protagonist will behave in the face of failure throughout the story. The protagonist of a story frequently encounters difficulties, and intention revision provides a systematic approach through which the character struggles to accomplish its goals. Both the BDI framework and intention revision contribute to the creation of suspense and are manipulated by the introduction of suspenseful action sequences, which are shown in the green box.

The IRIS planner contains a set of experimentally validated suspenseful action templates that it uses in the planning process. These templates can be translated into action sequences which are introduced to create the suspense in the story. Suspense is created by an antagonistic character taking an action which will thwart the protagonist’s goal. This malicious action, which
is known to the audience but unknown to the character, is afforded by the use of the BDI mental modeling discussed above. The audience feels suspense as the character unknowingly proceeds with its doomed plan. The suspense comes to a climax when the character realizes its failure and must perform intention revision to find another way to achieve its goal.

The output of the IRIS system is a story outline with suspenseful story moments, as represented by the dark blue box. This outline provides a skeleton of a story which can be instantiated into different media. In particular, the IRIS system has been shown to be effective at creating suspenseful, non-interactive text story fragments and a suspenseful, interactive text adventure game. The IRIS system is a useful tool for authors because it can provide a suspenseful basis of a story to which the author can then add his or her own personalization.

3.2 Definitions

The following definitions are used in the IRIS story generation algorithm. They define aspects of the BDI mental model that the protagonist uses as well as terms used in plan creation and intention revision.

Definition 3.1 A World Condition is a literal that describes a statement about the world.

Definition 3.2 An Action is a triple $< N, R, E >$ where $N$ is the action name, $R$ is the set of preconditions, which are world conditions that must be true to execute the action, and $E$ is the
set of effects, which are world conditions that will be made true after the successful execution
of the action.

Definition 3.3 A Plan is a pair \( P = <S, O> \) where \( S \) is a set of actions and \( O \) is a total
ordering over the actions.

Definition 3.4 For every plan \( P = <S, O> \), we use the name \( s_e \) to designate the last step in
\( S \) as determined by \( O \). \( s_e \) consists of a null action with no effects.

The first four definitions describe planning terms which have been adapted for the IRIS
system. World conditions are statements about the world. Actions describe transformation on
the world conditions, and a plan is a totally ordered set of actions with \( s_e \) as the last step. \( s_e \)
contains only preconditions and no effects. This step encodes the goal state of the plan.

Definition 3.5 An Intention is a triple \( <world\ condition, time, P> \), where \( P = <S, O> \)
and where \( time \) is the integer index in \( O \) of the step in plan \( P \) in which the world condition is
satisfied.

Intentions are world conditions that the character has committed to make true at a given point
in its plan.

Definition 3.6 Maintenance Intentions are intentions that are translated directly from the
set of world conditions that the author supplies as initial input to the system and that the author
desires to be satisfied by \( s_e \).

Maintenance intentions are intentions that need to be satisfied by the end of the plan. They are
are a subset of the preconditions of \( s_e \). If an action is selected in a plan that reverses a mainte-
nance intention that was previously satisfied, some other action must restore the maintenance
intention by the end of the plan.

Definition 3.7 Achievement Intentions are translated from a set of world conditions into
the preconditions of \( s_e \). These world conditions are provided by the author as initial input to
the system and represent a character’s goals. The must be established at some point in \( P \) but the
truth value does not need to be preserved all of the way to \( s_e \).

Achievement intentions are intentions that, once satisfied, do not need to remain satisfied. They
are are a subset of the effects of the actions in steps that come before the last step in \( P \). Other
subsequent actions can reverse achievement intentions.
Definition 3.8 **Temporary Intentions** are the set of intentions that consist of the preconditions of the actions in $P$.

Temporary intentions are named such since the agent is only weakly committed to them. If the action to which the temporary intentions are bound is removed from the plan, the temporary intentions will be dropped, since the character only adopted them to complete that action. Temporary intentions are only ever adopted to complete actions that work towards the fulfillment of achievement or maintenance intentions.

**Definition 3.9** A **Belief** is a literal.

**Definition 3.10** A **Desire** is a tuple $<l, v, s, k>$, where $l$ is a literal and $v$, $s$, and $k$ are labels such that $v \in \{\text{low, high}\}$, $s \in \{\text{character, author}\}$, $k \in \{\text{authorial, maintenance}\}$.

The low, high, character, and authorial labels allow for prioritization if the character has to choose between conflicting desires. The character prioritizes: high authorial, high character, then low character desires.

**Definition 3.11** A **Protagonist Character** $c$ is a 7-tuple $<N_c, B_c, D_c, I_c, \Lambda, P_c>$ where $N_c$ is a unique string of the character’s name, $B_c$ is the character’s set of beliefs, $D_c$ is the character’s set of desires, $I_c$ is the character’s set of all of its intentions, $\Lambda$ is the set of actions the character can perform, and $P_c$ is a plan that, when executed, makes the intentions in $I_c$ true.

**Definition 3.12** An **Authorial Goal** $G_A$ is a pair $<D_A, i_A>$, where $D_A$ is a set of desires denoting the conditions in the world the author wants to hold in the goal state, and $\forall d \in D_A$, $d$ is a high value, authorial desire. $i_A$ is a nonempty set of protagonist characters to which the authorial goal could be assigned.

These authorial goals are combined with a protagonist character’s desires to form the character’s initial intentions. When forming these intentions, it is possible that some of the intentions could be in conflict with each other. Since the authorial goals are given high value and authorial labels, they have higher priority over the character labeled intentions if there is a conflict between the two. Author-supplied goals are also given the maintenance label. This ensures that a character will satisfy goals that the author designates as important. More on resolving protagonist intention conflicts can be found in Section 3.5.7.
Definition 3.13 Given an action \( a \) and a protagonist character \( c = \langle N_c, B_c, D_c, I_c, \Lambda, P_c \rangle \), where \( P = \langle S, 0 \rangle \), a **Belief Update** is the modification of \( c \)'s beliefs \( B_c \) and the effects of \( a \) to produce an updated set of beliefs \( B_c \). The effects of \( a \) are joined with \( B_c \), and if any world conditions are in conflict, the world condition in the effects of \( a \) supersedes the conflicted world condition in \( B_c \).

Definition 3.14 **Intention Revision** is the addition or subtraction of a character’s intentions and/or the modification of the character’s plan.

Definition 3.15 **Working Assumptions** are the minimal set of world conditions that, when violated, will prompt intention revision. At a given time \( t \), the working assumptions consist of the preconditions of the actions of the unexecuted plan that are not satisfied in an effect of an action in the remaining plan.

An example of a character’s working assumptions is the following:

Given a character with the following actions and the actions’ preconditions and effects:

- **Action 1**: Preconditions: \( A, B \)  
  Effects: \( C, D \)
- **Action 2**: Preconditions: \( C, E \)  
  Effects: \( F, G \)
- **Action 3**: Preconditions: \( A, H \)  
  Effects: \( I, J \)

The character’s working assumptions are \( A, B, E, H \). Thus the character will perform intention revision if and only if the effect of the selected action violates any of these working assumptions. The character’s working assumptions are \( A, B, E, H \). Thus the character will perform intention revision if and only if the effect of the Drama Manager action violates any of these working assumptions.

Definition 3.16 **Projection** is an intention revision due to a belief update because the action \( a \) from the belief update has effects that violate the working assumptions.

Definition 3.17 **Reflection** is an intention revision due to a belief update because the action \( a \) from the belief update has preconditions are not part of the working assumptions.

Definition 3.18 **Optimization** is a intention revision where every action \( A \) is removed such that \( \forall e \in A, e \) is an effect of \( A \), \( e \in (\{ \text{All world conditions} \} - \{ \text{the preconditions of all subsequent actions} \}) \) || (the world conditions in the belief update).

Projection, reflection, and optimization are the three kind of intention revision. Projection occurs when the character, looking forward, sees that its plan will no longer succeed. Reflection
occurs when an action failed that a character expected to succeed. The character looks back and realizes that some of the conditions necessary for the completion of the plan which the character thought were true were actually false. Optimization occurs when the character removes extraneous actions from its plan. It removes all actions such that every effect of the action is either is already satisfied in the world or is satisfied in a later action in the character’s plan. This would make the action unnecessary.

### 3.3 Suspense Templates

The IRIS suspense templates were created to cover the different ways suspense can be created from IRIS’s definition of suspense. The templates were designed to create localized moments of suspense at the story-action level. Within the story-action level, the IRIS system creates suspense from protagonist plan failure. The suspense templates are described in detail in Section 3.5.4.

### 3.4 Drama Manager Actions

The Drama Manager actions allow the system to transform an action in the mutable protagonist’s plan to a story action in the immutable story plan. The Drama Manager actions consist of selecting a protagonist’s action, inserting a conflicting action into the protagonist’s plan, and informing the protagonist to change its mental model. The Drama Manager actions are described in detail in Section 3.5.5.

### 3.5 IRIS Generation Algorithm

#### 3.5.1 Overview

Figure 3.2 describes IRIS’s narrative generation process. The input for the algorithm is a set of characters $C$ containing a protagonist character with initial BDI mental model (initial beliefs and weighted desires) and a list of supporting characters with optional 'friend' or 'foe' labels, a set of authorial goals $G_A$, initial world conditions, and a set of actions $\Lambda$. The output is a totally ordered complete plan where all of the authorial goals are satisfied.

The narrative generator first takes in this initial information about the story. The protagonist character has a set of beliefs and desires. The supporting characters do not have this BDI model, but rather serve to enhance the suspense created around the protagonist. The authorial
Algorithm 1: IRIS-Narrative-Generation Algorithm

**Initialization:** Let \( P = \emptyset \). For the protagonist \( p \), \( I_p = \text{create-intentions}(D_p, G_A) \). \( G_S = G_A \cup \forall i \in I_p \text{ such that } i \text{ is a high value intention} \). \( g_c = \emptyset \).

1. **Protagonist Plan Creation:** If \( P_p = \emptyset \), \( P_p = \text{find-plan}(B_p, I_p, \Lambda) \). \( g_c \) is the temporally first \( g \in G_S \).

2. **Application of Suspense Templates:** If \( g_c \) does not have a suspense templates applied to it, identify the actions in \( P_p \) that accomplish \( g_c \). Choose one of these actions to threaten. Apply an applicable suspense template. Fill in the minimum number of variables in the template with consistent terms.

3. **Drama Manager Action Addition:** If next Drama Manager Action \( a \) is not a ground term (free of variables), fill in variables consistent with world conditions and character intentions. Add \( a \) to \( P \). If an effect of \( a \) is \( g_c \), remove \( g_c \) from \( G_S \) and \( g_c = \emptyset \).

4. **Belief Update:** Perform belief update based on the Drama Manager action.

5. **Intention Revision:** Call Intention-Revision\((p, a)\)

6. **Repair of Suspense Templates:** Update \( g_c \)'s template with consistent terms.

7. **Recursive invocation:** If \( G_S \neq \emptyset \), call IRIS-Narrative-Generation\((C, G_S, \Lambda)\).

**Figure 3.2** Algorithm 1: IRIS-Narrative-Generation Algorithm

goals are assigned to the protagonist and the protagonist forms its maximally-weighted consistent subset of intentions from its desires and assigned authorial goals. It now has a set of beliefs as a starting state and intentions for an ending state, so it uses these to form its initial plan.

The Drama Manager finds the next goal around which to create suspense. It applies suspense templates that threaten the completion of that goal. Then, the Drama Manager adds the next appropriate Drama Manager action to the story. The protagonist has the opportunity to update its beliefs and perform intention revision based on the effect of this action. Any Drama Manager actions added by the application of the suspense templates that were damaged by the selection of the last Drama Manager action are now repaired. Finally, if there are no more suspenseful goals to satisfy and the system is not in the middle of fulfilling a suspenseful goal, the algorithm returns the completed plan. Otherwise, the algorithm is called recursively.

### 3.5.2 IRIS’s Belief/Desire/Intention Framework (Initialization)

The story plan and current suspenseful goal are set to null. The authorial goals are given to the protagonist as \(< \text{literal}, \text{High, Authorial, Maintenance}> \) desires. The protagonist then forms its initial set of intentions. It finds the largest consistent subset of authorial goals to convert to intentions. Then adds to its intentions the largest subset of high value desires that are consistent with its existing intentions. Then it does the same for low value desires. Any
inconsistent desires are removed. The set of suspenseful goals $G_S$ is set equal to all of the High value intentions in the protagonist’s BDI model.

### 3.5.3 Protagonist Plan Creation (Step 1)

If the protagonist’s plan is null, the protagonist creates a plan using its BDI mental model. The beliefs serve as a start state, and its intentions as the goal state. The current suspenseful goal $g_c$ is set to the first goal in $G_S$ to be achieved by the protagonist’s plan. IRIS uses the partial order planning (POP) algorithm called Longbow [48]. Planning is discussed in detail in Section 2.5. 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.

### 3.5.4 Application of Suspense Templates (Step 2)

IRIS creates suspense from protagonist plan failure. The protagonist’s plan will fail if one of the preconditions of the actions in its plan does not hold. In this step, a threat by an antagonistic character is injected into the protagonist’s plan. There are three ways that the precondition can fail to hold.

The first way a precondition can fail to hold is an inconsistency between the protagonist’s beliefs and the actual state of the world. This inconsistency can be unknown to the audience too, so that the audience is surprised by the plan failure, or known to the audience, resulting in dramatic irony.

The second way a precondition can fail to hold is if the system deliberately modifies the state of the world to introduce plan failure. In this case, a precondition of the protagonist’s plan was true up to a point, but then was reversed by the system. This is analogous to initial state revision in planning [30]. This modification is unknown to the protagonist, but also unknown to the audience. It is kept secret from the audience because the change is a retroactive modification to the initial state of the world which seemingly has no perpetrator, so it would look unnatural for the audience if they saw it changed.

The third way a precondition can fail to hold is if an antagonistic agent takes action to thwart the precondition in the protagonist’s plan. This agent could be another character or nature. The action may or may not be observed by the protagonist. It is observed by the audience, because otherwise this condition would be a subset of the first condition where, unknown to
the protagonist and the audience, the state of the world is inconsistent with the protagonist’s beliefs.

The suspense templates were created from an enumeration of all of the Drama Manager actions that resulted in a protagonist plan failure. There were originally more Drama Manager actions and suspense templates (See Figure A.1), however, a number of these were not rated as suspenseful in the experimental evaluation in Chapter 5. The suspense templates listed in Figure 3.3 were those that were rated as suspenseful in the evaluation and subsequently integrated into the IRIS story generation algorithm.

If \( g_c \) does not have a suspense template assigned to it, then one needs to be assigned. A suspense template is applied to a goal as follows:

1) Identify the actions that are part of the plan to achieve \( g_c \). This can be done by using a simple algorithm that runs in \( O(n) \) that works backwards from the end of the plan and adds actions to a list that are needed to satisfy the preconditions of the final action or another action added to the list in this way.

2) Choose one of these actions non-deterministically, selecting from an action labeled by the author as conducive to suspense if possible.

3) Select a precondition of the chosen action to invalidate. To do this, the system first non-deterministically selects an antagonistic character. Then, for each precondition of the chosen action, the system tries to find a plan for the antagonistic character with the current state of the world as the start state and each precondition in turn as the goal state. Those preconditions without a resulting plan are eliminated from consideration.

4) For every precondition with an identifiable antagonistic plan, the system then sees if the protagonist could perform intention revision to reestablish the invalidated precondition. Those preconditions that the protagonist cannot reestablish are eliminated from consideration.

5) A precondition is selected non-deterministically from the remaining preconditions.

6) The antagonistic actions that invalidate this precondition need to be incorporated into a suspense template (see Figure 3.3). One of the applicable templates is chosen non-deterministically. Bindings in the template are filled in with a least-commitment approach.

7) The suspense template actions are inserted into the plan directly before the action that completes the suspenseful goal.

For an example of the application of a suspense template, see Section 3.6.
1. Conflicting Action a, Withhold True Observation of a
   Failed Action b
   True Observation of b

2. Conflicting Action a, Character not present for True Observation of a
   Failed Action b
   True Observation of b

3. True Utterance a
   Conflicting Action b, Withhold True Observation of b
   Failed Action g
   True Observation of g

4. True Observation a
   Conflicting Action b, Withhold True Observation of b
   Failed Action g
   True Observation of g

5. True Utterance a
   Conflicting Action b, Character not present for True Observation of b
   Failed Action g
   True Observation of g

6. True Observation a
   Conflicting Action b, Character not present for True Observation of b
   Failed Action g
   True Observation of g

**Figure 3.3** List of the validated IRIS suspense templates. These templates consist of a series of Drama Manager actions, that when inserted into the protagonist’s plan before a given protagonist goal, create dramatic irony and suspense around the goal. For more information on the Drama Manager actions, see Section 3.5.5.
3.5. Drama Manager Action Addition (Step 3)

The Drama Manager adds the next action from the mutable protagonist plan to the immutable story plan. If the next action contains ungrounded variables (meaning it is an action from a suspense template), the variables need to be grounded at this time.

The Drama Manager has the following available actions:

**Select Character Action(L)**

Description: “The Drama Manager selects an action from a character’s plan. The action happens at location L”

Preconditions:
- Preconditions of the action

Postconditions:
- Postconditions of the action
- For each effect E at L, True Observation(protagonist, E, L)

For this action, the Drama Manager simply selects an action for a character’s plan. In the way that IRIS is defined, only the singular protagonist will have a plan, so the protagonist will always be the one who’s plan is selected. This action is left in its general form if the system is ever extended to allow for multiple protagonists.

After the effects of the actions are updated in the world conditions, the protagonist observes all of the effects of the action as well.

**True Observation(A, N, X)**

Description: “A correctly observes N at location X”

Precondition:
- observableFrom(N,X)
- atLocation(A,X)
- Holds(N)
- Character A is the protagonist

Postcondition:
- Believes(A,N)

For this action, the protagonist observes some true state of the world that is observable from its location. The protagonist then adopts this world condition in its beliefs.
3.5. IRIS GENERATION ALGORITHM  CHAPTER 3. IRIS GENERATION PROCESS

**True Utterance(A, B, N, X)**

Description: “A truthfully informs B of N at location X”

Precondition:
- Character A,
- Character B,
- atLocation(A, X),
- atLocation(B, X),
- Character B is the protagonist
- Believes(A, N)

Postcondition:
- Believes(B, N)

For this action, one character conveys a belief that it holds to the protagonist that is present at the same location. The protagonist then adopts this world condition in its beliefs.

**Conflicting Action (L)**

Description: “An antagonistic character takes an action that will thwart a precondition of a step in the protagonist’s plan.”

Preconditions:
- Preconditions of the action
  - There exists an effect $e$ of the action such that there exists a precondition of an action in the protagonist’s plan $\sim e$.

Postconditions:
- Postconditions of the action

An antagonistic character takes an action that has an effect which thwarts a precondition in the protagonist’s plan.

3.5.6 Belief Update (Step 4)

If a Select Character Action was chosen in step 3, then the protagonist’s beliefs are updated with the effects of the action that was selected.
3.5. IRIS GENERATION ALGORITHM  CHAPTER 3. IRIS GENERATION PROCESS

Algorithm 2: Intention-Revision (c, v)
1. **Working Assumptions Violation Check**: See if any of the effects of the chosen action \(a\) violate the working assumption of the character \(c\).
2a. **Optimization Prune**: If \(c\) did not have a working assumption violation, remove each action \(a'\) that is an element in \(c\)’s plan \(p\), such that \(\forall e \in a', e\) is an effect of \(a', e \in \{\text{All world conditions}\} - \{\text{the preconditions of all subsequent actions in } p\}\) \(\triangledown\) (the belief update from the effects of selected action \(a\)). Return.
2b. **Replan With New Beliefs**: If \(c\) did have a working assumption violation, \(P_c = \text{find-plan}(c, B_c, I_c, \Lambda)\)
3b. **Drop Intentions Until a Working Plan Can be Found**: If \(P_c = \emptyset\), \(P_c = \text{find-plan}(c, B_c, I_c, \Lambda)\). If \(P_c\) still = \(\emptyset\), \(I_c = \text{find-max-consistent-subset}(I_c)\), then repeat this step.

Figure 3.4 Algorithm 2: Intention-Revision Algorithm

3.5.7 Intention Revision (Step 5)

Based on the effects of the selected action, the protagonist might need to perform intention revision (See Figure 3.4).

Intention revision is the addition or subtraction of a character’s intentions and/or the modification of the character’s plan. Intention revision is not to be confused with plan revision. Plan revision is only an alteration of the actions in a character’s plan, without considering its intentions. Plan revision is an aspect of intention revision, though. Intention revision consists of:

**Working Assumptions Violation Check**: See if any of the effects of the selected action violate the working assumption of the protagonist.

**Optimization Prune**: If the protagonist did not have a working assumption violation, then it does not need to replan or modify its intentions. However, it is possible that some of the actions in its plan are now unnecessary due to the effects of the selected action. For example, perhaps the character planned to sneak up to the guard, steal his keys, unlock the door, and walk through. The selected action caused an explosion that destroyed the door. It is now unnecessary to perform the first three steps in the character’s plan, since it can now just walk through the door. The first three steps should be culled. This is done by removing all actions in the character’s plan whose only effects are already satisfied or were just satisfied by the effects of the action. After this optimization prune, go to step 6 of Algorithm 1.

**Replan With New Beliefs**: If the protagonist did have a working assumption violation, then it needs to repair its plan, intentions, or both. First, try to replan with the same intentions as the goal state and its updated set of beliefs as the start state.
3.6 Example Story Generation

**Drop Intentions Until a Working Plan Can be Found:** If replanning alone did not produce a plan to satisfy all of its intentions, some intentions need to be dropped. A minimally weighted, inconsistent subset of intentions is dropped until a plan can be found.

It is always possible to know if the character will perform intention revision if a given action is selected in Step 3 of Algorithm 1. This is due to the fact that IRIS tracks the character’s working assumptions. The working assumptions are the minimal set of world conditions that, when violated, will prompt intention revision. For an example of working assumptions and how they can be violated, see Definition 3.15.

### 3.5.8 Repair of Suspense Templates (Step 6)

It is possible that the effects of the chosen action made it such that the actions of the suspense template are now inconsistent with the world conditions (for example, if a character is expected in one of these actions but is now dead). If this is the case, repair these actions by choosing new ground terms consistent with the world conditions.

### 3.5.9 Recursive invocation (Step 7)

If there are still more suspenseful goals unsatisfied, call the algorithm recursively.

### 3.6 Example Story Generation

This is an example fragment of story generation that the IRIS system produced in the Western domain (See Appendix C). The current suspenseful goal is \((\text{has PLAYER DYNAMITE})\). The relevant portion of the character’s plan is shown in Figure 3.5.

A suspense template needs to be applied to this goal. The system first finds the actions that contribute to the goal of \((\text{has PLAYER DYNAMITE})\). In this case, it is both \((\text{walk PLAYER MAIN-STREET MINE NODOOR})\) and \((\text{pickup PLAYER DYNAMITE MINE})\). \((\text{talk PLAYER RANCHER MINE})\) is an action that satisfies another goal, so it is not considered. Next, the system selects one of these actions to threaten. Since in the Western domain the pickup action is label as suspense-conducive and walk is not, pickup is selected. If there were multiple suspense-conducive choices, then one would have been selected non-deterministically.

The system now needs to select a precondition of the pickup action from the protagonist’s plan to invalidate. This binding of the \((\text{pickup PLAYER DYNAMITE MINE})\) action the five pre-conditions shown in Figure 3.6.
3.6. EXAMPLE STORY GENERATION  CHAPTER 3.  IRIS GENERATION PROCESS

Next the system chooses an antagonistic character to take a series of actions that will invalidate one of these preconditions. The MINER character is selected non-deterministically from the available choices. Given the current state of the world as the start state and each of the preconditions in turn as the end state, the system tries to find a plan for the miner that invalidates the given precondition. There is no set of actions that can result in \((\text{not } \text{item DYNAMITE})\), so that precondition is eliminated from consideration. \((\text{not } \text{at PLAYER MINE})\) is already true since the player is currently at the main street, so this condition already holds and is eliminated. The remaining three preconditions do have associated plans that can be found, as shown in Figure 3.7.

The system then sees if the protagonist, through intention revision, would be able to achieve the suspenseful of \((\text{has PLAYER DYNAMITE})\). If the \((\text{not } \text{alive PLAYER})\) precondition is invalidated, the protagonist will not be able to perform intention revision to fulfill the
3.6. EXAMPLE STORY GENERATION

CHAPTER 3. IRIS GENERATION PROCESS

Figure 3.6 The preconditions for the suspenseful actions being threatened. An antagonistic character will take a series of actions that threatens the completion of one of these preconditions.

(has PLAYER DYNAMITE). Therefore (not (alive PLAYER)) is eliminated from consideration. For the other two preconditions, Figure 3.8 shows the plans that the protagonist could adopt to complete the suspenseful goal.

Therefore both of these preconditions are eligible to be invalidate by an antagonistic character’s plan and then subsequently repaired. One precondition is selected non-deterministically, and for this example we will say (not (at DYNAMITE MINE)) is selected.

The conflicting action sequence now needs to be integrated into a suspense template. The subset of available suspense templates that apply are:

1) Conflicting Action((pickup MINER DYNAMITE MINE))
(withhold True Observation of (has MINER DYNAMITE))
Failed Action((pickup PLAYER DYNAMITE MINE))
True Observation(has (MINER DYNAMITE))

2) True Utterance(?FRIENDLY-CHARACTER, (at DYNAMITE MINE)))
Conflicting Action((pickup MINER DYNAMITE MINE))
(withhold True Observation of (has MINER DYNAMITE))
Failed Action((pickup PLAYER DYNAMITE MINE))
True Observation(has (MINER DYNAMITE))

One of these is selected non-deterministically, say the first. These actions are inserted into the plan immediately before the (pickup PLAYER DYNAMITE MINE) action. The actions currently
Figure 3.7 Antagonistic plans that can be generated to invalidate the different preconditions of the suspenseful action.

in the plan are shown in Figure 3.9.

The first four actions in the plan are selected one at a time, removed from the plan, and added to the story. After the Failed Action (pickup PLAYER DYNAMITE MINE), the protagonist performs a Reflection intention revision and realizes that the (at DYNAMITE MINE) condition for its action did not hold. It also notices that it can retain the (has PLAYER DYNAMITE) goal if it modifies its plan. This was anticipated earlier when the (at DYNAMITE MINE) precondition was being selected to be threatened. The character replans and the plan is shown in Figure 3.10.

These actions are successfully executed and the suspenseful goal of (has PLAYER DYNAMITE) is completed. The process now continues for the next suspenseful goal.

The complete set of story actions for this suspenseful goal is shown in Figure 3.11.

A textual description of the events might be “The player wants to get the dynamite from the mine. However, the miner who is in the mine grabs the dynamite. The player moves to the mine, searches for the dynamite and can’t find it. The player then realizes that the miner has the dynamite, threatens the miner with a knife, and steals it from the miner.”
Figure 3.8 Different plans the protagonist can adopt to repair the violated preconditions of the suspenseful actions.
Figure 3.9 The current protagonist plan after the suspenseful actions sequence is inserted.
Figure 3.10 The protagonist plan after the character performs intention revision due to a failed action.
Figure 3.11 The complete set of actions for the suspenseful goal.
Chapter 4

Non-Interactive Evaluation

4.1 Introduction

In Chapter 3, I discussed the planning algorithm that IRIS uses to create suspenseful story outlines. In this chapter, I address the question,

“Does the output of IRIS system create suspense in a reader in the context of a non-interactive text story?”

Over the course of two experiments, four different suspense creation processes were compared: the IRIS suspense templates, an operationalization of a case-based reasoning system with narrative properties called MINSTREL [41], a human author with creative writing experience, and a logically and temporally consistent random baseline. Participants evaluated pair-wise comparisons of the different processes with regards to suspense. The stories produced by the human author and by most of the IRIS fragments, particularly those involving a disparity of knowledge between the audience and the protagonist, were significantly more suspenseful than the random story fragments. Additionally, the IRIS suspense templates did not perform significantly differently than the human author. This shows that the output of the IRIS system can create suspense in a reader in the context of a non-interactive text story.

4.2 Initial Experiment

An initial experiment evaluated the IRIS templates against multiple other suspense creation processes: three human authors, an operationalization the MINSTREL system, and a logically and temporally consistent random baseline. The hypotheses were that the human authors and
the IRIS fragments would be rated higher than the random baseline, and that the IRIS fragments would be rated comparably to the human authors. This rating would be done by participants reading the fragments and selecting which fragment, if any, was more suspenseful. It was found that a problem in the story domain and fragment creation rules may have disrupted the results of the experiment, necessitating a follow-up experiment with a different domain and set of story fragment creation rules.

4.2.1 Experimental Design

In the initial experiment, the story creation processes were given a bank of English sentences to use in story fragment creation (See Appendix A). The domain was a medieval setting where a knight trying to slay a dragon. Example sentences included “The knight attacks the dragon,” and “The wizard tells the knight where the dragon is hiding.” Appendix A also discusses the rules were given to ensure that the story would be logically and temporally consistent. This was done to ensure that that participants did not rate the fragment poorly because it did not make sense.

The study was conducted online in HTML with JavaScript to help handle the selection and display of the story fragments. Participants were recruited by snowball sampling through email, message boards, in person, and on Facebook. The participants were first given a link to the consent form. Then, they were given a demographic survey asking their age and gender. In the demographic survey and the comparisons, the participants always had the option to leave any question unanswered. Next, they were given the background information about the domain of the story.

The participants were then given a series of sentence fragment pairs. They were asked to read each fragment and compare which was more suspenseful. They were given four options: Story Fragment A was suspenseful and Story Fragment B was not, Story Fragment B was suspenseful and Story Fragment A was not, Both were suspenseful, or Neither were suspenseful. After selecting their choice hitting the Next button, they were shown another pairing.

Four hypotheses were tested:

**Hypothesis 1: The story fragments generated with the IRIS suspense templates will be comparably suspenseful to the fragments generated by the human authors selecting from the same set of story actions.**

To ensure that the story fragments that the suspense templates and the human authors generate are able to be compared, they both compose their fragments from the same set of story actions.
This puts a limitation on the human author since he or she can only use a limited set of story actions instead of an unlimited vocabulary. If this hypothesis holds, then the suspense templates will have been shown to be sufficient to generate suspense at the story-action level comparable to a human author selecting from the same set of story actions.

**Hypothesis 2:** The story fragments generated with the IRIS suspense templates will be more suspenseful than the randomly-generated baseline.

It needs to be shown that the suspense templates are an improvement over no method of creating suspense at all.

**Hypothesis 3:** The story fragments generated by the human authors will be more suspenseful than the randomly-generated baseline.

This demonstrates that the human authors were actually able to create more suspense than random combinations of story events.

**Hypothesis 4:** The story fragments generated with the IRIS suspense templates will be at least as suspenseful as the MINSTREL formula.

Since the IRIS suspense templates are more complicated to implement and apply than the MINSTREL formula, it is desirable that they perform at least as well as the simpler MINSTREL approach.

### 4.2.2 Results

117 people completed the demographic survey and were presented with the experiment. There were 34 women with an average age of 33.6 +/- 15.2, and 83 men, with an average age of 26.2 +/- 9.8.

Table 4.1 and Figure 4.1 show the results of the comparisons with the IRIS suspense templates. Using a series of one-tailed, 2x2 Chi-Squared tests without Yates' corrections, the IRIS templates never significantly outperformed the random baseline. Some templates performed poorly against all four of the other fragment creation processes. This is significant because they were included in the follow-up experiment and also performed poorly, whereas the templates with marginal results in the initial experiment performed much better in the follow-up experiment. This suggests inter-experiment validity because the templates that did poorly in the first
Figure 4.1 A visualization of the comparison between the IRIS suspense templates and the three other creation processes in the initial experiment. Red indicates that the column was rated higher and black that they were not significantly different, both at $p = 0.05$ level. Gray indicates that more than $1/3$ of the responses rated neither as suspenseful.
Table 4.1 p values for IRIS suspense templates versus the five other creation processes in the initial experiment. A one-tailed Chi-Squared test was performed versus the random baseline, and a two-tailed Chi-Squared test was performed versus the others. A dash indicates that more than 1/3 of the responses rated neither as suspenseful.

<table>
<thead>
<tr>
<th>Template</th>
<th>Random</th>
<th>MINSTREL</th>
<th>Human 1</th>
<th>Human 2</th>
<th>Human 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Template 1</td>
<td>0.03</td>
<td>0.03</td>
<td>0.05</td>
<td>0.18</td>
<td>0.03</td>
</tr>
<tr>
<td>Template 3</td>
<td>-</td>
<td>0.09</td>
<td>0.06</td>
<td>0.37</td>
<td>0.09</td>
</tr>
<tr>
<td>Template 4</td>
<td>-</td>
<td>-</td>
<td>0.11</td>
<td>0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>Template 5</td>
<td>0.11</td>
<td>-</td>
<td>0.01</td>
<td>-</td>
<td>0.001</td>
</tr>
<tr>
<td>Template 6</td>
<td>0.3</td>
<td>0.001</td>
<td>0.05</td>
<td>0.001</td>
<td>0.03</td>
</tr>
<tr>
<td>Template 7</td>
<td>0.24</td>
<td>0.87</td>
<td>0.55</td>
<td>0.05</td>
<td>0.3</td>
</tr>
<tr>
<td>Template 8</td>
<td>0.36</td>
<td>1</td>
<td>0.11</td>
<td>0.06</td>
<td>0.05</td>
</tr>
<tr>
<td>Template 9</td>
<td>0.18</td>
<td>0.66</td>
<td>0.51</td>
<td>0.06</td>
<td>0.29</td>
</tr>
<tr>
<td>Template 10</td>
<td>-</td>
<td>-</td>
<td>0.54</td>
<td>-</td>
<td>0.002</td>
</tr>
<tr>
<td>Template 11</td>
<td>0.44</td>
<td>-</td>
<td>0.73</td>
<td>0.57</td>
<td>0.43</td>
</tr>
</tbody>
</table>

also did poorly in the second. If they had seen a significant improvement, it would have been less likely that the sentence creation in the initial experiment was confounding the performance of the IRIS suspense templates.

The other fragment creation processes were also compared to the random baseline. The null hypothesis for these comparisons was that they would perform comparably to random. The alternative hypothesis was that they would perform better than random. A one-tailed, 2x2 Chi-Squared test without Yates’ correction was performed for human authors 2 ($\chi^2(1) = 0.34, p = 0.28$) and 3 ($\chi^2(1) = 3.00, p = 0.06$). Human author 3 seemed to perform better than random, but human author 2 did not. For human author 1 and MINSTREL, the participants reported more counts of preferring random than these methods, so a two-tailed, 2x2 Chi-Squared test was performed for human author 1 ($\chi^2(1) = 4.06, p = 0.04$) and MINSTREL ($\chi^2(1) = 0.85, p = 0.89$). This indicates that these methods did not perform better than random; human author 1 actually performed worse than random.

### 4.2.3 Discussion

The results of this initial experiment showed that all creation processes, with the exception of one of the human authors, were rated similarly with regard to suspense. However, few of the fragments were rated as unsuspenseful. It was hypothesized that something other than the creation process was introducing suspense, i.e., the sentences themselves. In the follow-
up experiment, the story actions were constructed in a template-based way instead of the sentence structure of the first experiment, to mitigate the effect of discourse-level suspense being introduced into the sentences themselves.

Another interesting result was that about halfway through the initial experiment, it was observed that the one sentence fragment stories were rated significantly lower than fragments with two or more sentences. A one-tailed 2x2 Chi-squared test testing if the one sentence fragment were rated lower than those of two or more was significant ($\chi^2(1) = 38.05, p < 0.0001$). There was no significant difference among fragments with regards to length as long as they had two or more sentences. The initial set of IRIS templates had several one sentence templates (like template 2). These were removed from the follow-up experiment.

Finally, the participants were given 30 pair-wise comparisons to make among the different story fragments. There was some concern that they may experience fatigue partway through the experiment and begin to answer the questions differently. Pair-wise 2x2 two-tailed Chi-squared tests were run on each pairing of the 30 questions. No significant difference was found for the responses as the survey progresses.

4.3 Follow-Up Experiment

In the follow-up experiment, the same suspense creation processes were compared with the same hypotheses. However, the domain and story actions were changed to minimize the discourse level suspense that may have confounded the results of the initial experiment. It was found that for most of the suspense templates, particularly those involving a disparity of knowledge between the reader and the protagonist, the fragments produced by the IRIS templates were rated significantly more suspenseful than those produced by the random and MINSTREL templates and comparably to those produced by the human author.

4.3.1 Experimental Design

All of the different creation methods being tested produced story fragments using the same fragment creation grammar. The grammar is shown in Appendix A. The domain was a murder mystery. In this story, there is a detective that is trying to figure out who killed the host of a dinner party. This is the goal that all of the story fragments will try to create suspense around.

To prevent the possible introduction of the discourse-level suspense that proved problematic in the initial experiment, a grammar was used for the creation of story fragments. Fragment length was limited to range between 2 and 6 sentences in order to prevent readers from potential
distraction regarding the suspenseful goals. Additionally, the same person or place was not
allowed to be used for both parts of a fragment. For example “The maid attacks the maid” was
not allowed.

A human author with creative writing experience was recruited to produce suspenseful story
fragments. He was given the creation grammar, the rules described in the previous paragraph,
and told to create as many suspenseful story fragments around the goal of the detective trying
to find the killer as he felt he could generate. He produced seven story fragments between
lengths four and six sentences.

Next, a set of story fragments were created using the MINSTREL method of creating sus-
pense (see Section 2.7). Since IRIS is designed to create suspense at the story-action level and
not the discourse level, the step where emotion is interjected at the discourse level was omitted.
An example MINSTREL fragment was “The maid attacked the detective. The detective tried
to move to the attic but failed.”

Then, story fragments from the IRIS suspense formulas were created. There were multiple
ways to create fragments from each of the suspense formulae, so all of the combinations were
generated. There is some subtlety in templates 3, 7, & 8, and 4, 9, & 10. In the first three, an
action is performed without the main character’s knowledge, and in the second three an action
is performed without that character being present. However, the suspense is being generated at
the story-action level and not the discourse level, so the templates do not make any commitment
as to how this action is conveyed to the audience. Since the participants will be reading the
story as text and not LISP-style action syntax, a thin but necessary discourse is introduced.
These are the template-based “Modifiers” described in the template grammar in Figure A.5.

For the logically and temporally consistent baseline, a program was created that generated
random story fragments following the story creation rules. The actual story fragments were
generated at run time during each participant’s experimental session.

The study was conducted online in HTML with JavaScript to help handle the selection
and display of the story fragments. Participants were recruited by snowball sampling in email
and in person, and through undergraduate and graduate computer science classes. For students
recruited through courses, participation fulfilled an experimental course requirement. The par-
ticipants were first given a link to the consent form. Then, they were given a demographic survey
asking their age and gender. In the demographic survey and the comparisons, the participants
always had the option to leave any question unanswered. Next, they were given the background
information about the domain of the story.

The participants were then given a series of sentence fragment pairs. They were asked to
read each fragment and indicate which was more suspenseful. They were given four options:
4.3. FOLLOW-UP EXPERIMENT  CHAPTER 4.  NON-INTERACTIVE EVALUATION

Story Fragment A was suspenseful and Story Fragment B was not, Story Fragment B was suspenseful and Story Fragment A was not, Both were suspenseful, or Neither were suspenseful. After selecting their choice and hitting the Next button, they were shown another pairing. A screenshot of the survey page is shown in Appendix A.

There were 36 pair-wise combinations: 11 IRIS action templates x 3 other story creation methods + the 3 combinations of these 3 methods. At run time, a random fragment from the possible ones of the given approach was selected.

The same hypotheses were tested as in the previous experiment.

4.3.2 Results

83 people completed the demographic survey and were presented with the experiment, with an average age of 22.1 +/- 2.7. There were 9 women and 74 men who participated in the experiment. There were too few women to separate them out and compare their responses to those of the men, so this is a potential threat to external validity. A sizable portion of the participants were computer science undergraduates, which accounts for the low average age. To evaluate if the IRIS templates would be rated comparably to the human authors and if the templates would be rated comparably to MINSTREL, I used the two-tailed 2x2 Chi-square test without Yates’ correction to compare the participants’ response to a null hypothesis of equal responses for both approaches. The responses for “Both were suspenseful” were added to the counts for believing one method or the other was more suspenseful. For example, 32 people reported that they found template 7 more suspenseful than MINSTREL, seven found MINSTREL more suspenseful than template 7, 12 reported them both suspenseful, and 12 reported neither suspenseful. So the 12 that reported both were suspenseful were added to the counts for each of the two approaches, bringing the count up to 44 for template 7 and 19 for MINSTREL. This was compared with a null hypothesis of 31 for each, which is half of the responses going to each approach (Chi-square tests can’t use decimal numbers). If there was no difference in the two methods, then the reported counts would not differ significantly from the expected results. In this case, $\chi^2(1) = 5.13$, $p = 0.02$, indicating that the participants report template 7 more suspenseful than MINSTREL.

To evaluate Hypothesis 2, that the IRIS templates would be rated higher than random, and Hypothesis 3, that the human authors would be rated higher than random, we used the one-tailed 2x2 Chi-square test without Yates’ correction to compare the participants’ response to a null hypothesis of equal responses for both approaches. Here the process was similar to before, except that the responses “Both were suspenseful” were discarded since we wanted to
Figure 4.2 A visualization of the comparison between the IRIS suspense templates and the three other creation processes in the follow up experiment. Red indicates the column was rated higher, green that the row was rated higher, black that they were not significantly different, all at p = 0.05 level. Gray indicates that more than 1/3 of the responses rated neither as suspenseful. One comparison was right at this 1/3 but had a highly significant p value.
### Table 4.2

$p$ values for IRIS suspense templates versus the three other creation processes in the follow-up experiment. A one-tailed Chi-Squared test was performed versus the random baseline, and a two-tailed Chi-Squared test was performed versus the other two. A dash indicates that more than 1/3 of the responses rated neither as suspenseful.

<table>
<thead>
<tr>
<th>Template</th>
<th>Random</th>
<th>MINSTREL</th>
<th>Human</th>
</tr>
</thead>
<tbody>
<tr>
<td>Template 1</td>
<td>-</td>
<td>0.46</td>
<td>0.06</td>
</tr>
<tr>
<td>Template 2</td>
<td>-</td>
<td>-</td>
<td>0.0001</td>
</tr>
<tr>
<td>Template 3</td>
<td>0.001</td>
<td>0.01</td>
<td>0.47</td>
</tr>
<tr>
<td>Template 4</td>
<td>0.003</td>
<td>0.05</td>
<td>0.92</td>
</tr>
<tr>
<td>Template 5</td>
<td>0.35</td>
<td>0.44</td>
<td>0.12</td>
</tr>
<tr>
<td>Template 6</td>
<td>0.34</td>
<td>0.52</td>
<td>0.09</td>
</tr>
<tr>
<td>Template 7</td>
<td>0.002</td>
<td>0.02</td>
<td>0.10</td>
</tr>
<tr>
<td>Template 8</td>
<td>0.001</td>
<td>0.05</td>
<td>0.39</td>
</tr>
<tr>
<td>Template 9</td>
<td>0.13</td>
<td>0.04</td>
<td>0.52</td>
</tr>
<tr>
<td>Template 10</td>
<td>0.04</td>
<td>0.17</td>
<td>-</td>
</tr>
<tr>
<td>Template 11</td>
<td>-</td>
<td>-</td>
<td>0.51</td>
</tr>
</tbody>
</table>

measure if one performed higher, not just comparable.

In all cases, if more than 1/3 of the responses were “Neither were suspenseful,” then the above tests were not performed. This is a threshold we decided on to indicate that the participants found both approaches to lack suspense.

One-tailed 2x2 Chi-Squared tests were performed to measure if MINSTREL and the human author performed significantly better than random. The test for MINSTREL failed to reject the null hypothesis of performing comparably to the random baseline, ($\chi^2(1) = 0.01$, $p = 0.45$), but the test for the human author rejects the null hypothesis of performing comparably to the random, ($\chi^2(1) = 2.33$, $p = 0.06$).

Table 4.2 and Figure 4.2 show the results of the follow-up experiment.

#### 4.3.3 Discussion

Hypothesis 3, that the human author would perform better than random, was supported. This is important because a human author is often considered the ideal that story generation systems strive towards, and it needs to be shown that the human author performed well enough to make useful claims when comparing a generation system to the human author.

Hypothesis 4, that the IRIS templates would perform at least as well as MINSTREL, was also supported. The IRIS fragments performed much better than MINSTREL, and MINSTREL did
not perform significantly differently than random. This could be due to the fact that MINSTREL requires some discourse level suspense that was purposively minimized in the experiment.

For six of the 11 IRIS templates, Hypotheses 1 and 2 were proven as well. The exceptions were templates 1, 2, 5, 6, and 11. These templates were removed from the IRIS generation algorithm. The most successful templates were those where the audience had knowledge about the protagonist’s imminent plan failure and the protagonist did not. This makes intuitive sense and matches the system’s definition of suspense which involves anxiety and uncertainty around the success of an outcome from the audience’s perspective.

4.4 Conclusion

In this experiment, I evaluated the efficacy of the IRIS system at generating suspense in the context of a non-interactive text story. Over the course of two experiments, four different suspense creation processes were compared: the IRIS suspense templates, an operationalization of a case-based reasoning system with narrative properties called MINSTREL, a human author with creative writing experience, and a logically consistent random baseline. Participants evaluated pair-wise comparisons of the different processes with regards to suspense. The stories produced by the human author and by most of the IRIS fragments, particularly those involving a disparity of knowledge between the audience and the protagonist, were significantly more suspenseful than the random story fragments. Additionally, the IRIS suspense templates did not perform significantly differently than the human author. These results show that it is possible to create localized, story-action level moments of suspense in a story using a small set of action templates comparable to a human author selecting from the same set of story actions.
Chapter 5

Interactive Evaluation

5.1 Introduction

In Chapter 4, I validated the IRIS system’s ability to create suspense in the domain of non-interactive text stories. In my thesis, I also claim that the IRIS system can create suspense in an interactive domain. In this chapter, I address the question,

“Does the output of IRIS system create suspense in a player in the context of an interactive text game? What, if any, considerations can be made to adapt the IRIS story output to an interactive context?”

Two transformations were proposed to be necessary when translating the IRIS story output to the domain of an interactive text game: players need to be informed of ways to replan towards their goals in the case of plan failure and restrictions need to be placed on player actions ensuring that they do not bypass suspenseful story moments. The chosen method of performing of these transformations was 1) an in-game method of giving hints of how to achieve the game’s goals and 2) the presence of the suspenseful action sequences from the original story outline and the restriction of player agency during their execution. I hypothesized that both of these transformations were necessary to preserve IRIS’s sense of suspense in an interactive context.

In this experiment, participants played one of four versions of a text adventure video game in which one, both, or neither of the transformations were used. After the game, participants answered survey questions that were create from the operationalization of IRIS’s definition of suspense. These questions measured participants’ sense of suspense during the execution of the three suspenseful goals from the story outline and several other non-essential, non-suspenseful
goals. It was found that the version of the game containing both transformations was rated higher in suspense than each of the three other versions of the game for two of the three suspenseful goals. Also, for the second transformation, ensuring that the players do not bypass the suspenseful story moments, the forced encounter of the action sequence was more important than the restriction of player actions. This validates IRIS as a viable tool in creating suspense in an interactive text adventure game when both of the transformations are applied.

5.2 Transitioning Contexts

The translation from a non-interactive story outline to an interactive experience that preserves the story outline’s sense of suspense is difficult for at least two reasons. First, the simplifying assumption in IRIS where the audience knows the complete state of the world at all times does not translate to an interactive narrative environment. Second, if given the choice, players in an interactive environment might knowingly or unknowingly take actions that interfere with the suspenseful situations that the author created in the outline.

However, there is support that such a translation can be made. In non-interactive contexts, suspense is largely created around the protagonist, who is at the forefront of the story. In an interactive game, the player will serve as the protagonist, so any suspense created will be experienced by him or her. Also, the player’s role in the creation of an interactive experience is analogous to an author’s role in the creation of a non-interactive narrative. Both can create a rich narrative experience by exercising authorial creativity with regard to action ordering, addition of narratively interesting but non-essential actions, and discourse choices while still adhering to restrictions placed on them by a set of rules from an external system.

There are two considerations that can be made when translating from the non-interactive outline to an interactive game in order to preserve suspense. The first consideration is that when the player’s plan fails, they need to have sufficient information about how to replan to achieve their goal. This ensures that players will not abandon plans around which suspense is being created. One important component of a helpful hint requires modeling the player’s knowledge. Modeling player knowledge has been shown to be important when providing dynamic feedback to the player [35]. Instead of having complete knowledge of the world at every point in the story, the player’s knowledge can be modeled as containing any introductory information given prior to the story, plus the effects of every action that takes place in the player’s location. This is a simplifying assumption since it assumes the player knows the effects of every action and information is uniformly salient. Since the player’s knowledge is modeled, the system will know what new information it needs to introduce at a given point in the story to allow the player
to continue on a new plan. Assumptions include that the player will know how to use the new information and that there is a non-immersion breaking way to introduce the information to the player.

The player needs to have a level of agency in the game to create the interactive experience, but at certain points in the story their level of control needs to be reduced to ensure that he or she experiences the suspense that the system is trying to create. Moderating perceived and actual player agency has an important effect on the player’s game experience [14, 27, 40]. The player should not be able to take actions that will bypass the suspenseful action sequences in the story. Small details, for example the actors or location of the event, could be altered, but the actions themselves and their ordering need to be preserved. This may require the system to prevent the player from performing an action if it would bypass the suspenseful situation or to guide the player to take actions that will trigger the suspense. Also, during the execution of the suspenseful sequence of actions, the player needs to have reduced agency with respect to the actions that can be performed. The player should not be allowed to walk away or take extraneous actions during these suspenseful moments. This restriction will be modeled as a dynamic list of actions available to the player that will be shown in the game interface.

5.3 Western Heist Game Domain

I provided the IRIS system with a western heist domain where the protagonist is a bank robber (See Appendix C). A visual representation of the game world is shown in Figure 5.1. Actions allowed in the domain included move, talk, pick up, give, attack, steal, unlock, blow up, and capture. The important items to gather were the key and the dynamite, which allowed the protagonist to make it past the two locked doors on the way to the treasure room. These items were available at several different places in the world. Once in the treasure room, the protagonist could take the loot and try to leave town. There were numerous characters in the world that could aid or hinder the protagonist and some characters present just to provide background flavor for the town.

The protagonist’s objective is to get the loot out of the vault. The IRIS system identified subgoals in the story where suspense should be introduced. The subgoals are 1) getting the dynamite used to blow up the vault door, 2) getting the key used to open the vault safe, and 3) moving the loot from the vault to the train. There were multiple ways to accomplish these subgoals. To get the dynamite, the player could convince the store owner to give him or her a stick or dynamite, or it could be found when exploring the mines. To get the key, the player could steal it from the bank teller or sneak into his room and steal it. When the player gets the
loot, another bandit steals it from you. To get it back, the player can attack the bandit or try to negotiate.

5.4 Experimental Design

This story outline was implemented into a single player Multi User Dungeon (MUD) environment called LambdaMOO. In the game, a participant plays as the bank robber with the objectives described above. A GUI was developed in C# to aid in game play (see Figure 5.2). The GUI presents a restricted subset of available player actions at a given time, as per transformation 2 described above. The order in which the actions were presented was randomized after every action selection to prevent priming bias.

Participants were recruited from Computer Science undergraduate classes. There were activities that the students could perform to receive extra credit in the class, and this experiment was one option. The participants came to the researcher’s lab for the study where they were first given a consent form to read and sign. Next, they played the game. After the game was over, they were given a survey to measure their sense of suspense around each available game goal. These goals included the three suspenseful goals as identified by IRIS and several others.
that were not targeted as suspenseful but were made available to increase the number of things
to do in the world. Participants rated their agreement on a five point Likert scale with the five
survey questions that were created from the operationalization of IRIS’s definition of suspense
(See Appendix B). For the survey, the order of the questions regarding the game goals and the
questions within each game goal were randomized for each participant.

A central hypothesis, based on the two proposed transformations described above, was
tested:

**Hypothesis**  Both 1) an in-game method of giving hints about the ways to achieve the game’s
goals and 2) the presence of the suspenseful action sequences from IRIS’s original story
outline and the restriction of player agency during their execution, are necessary to create
suspense in an interactive text adventure implemented using IRIS’s narrative output.

Four different versions of the game, using none, one, or both of the transformations, were
created. Participants were randomly assigned to the different versions. Version 1 contained
no game hints or suspenseful sequences, version 2 contained game hints but no suspenseful
sequences, version 3 contained no hints but suspenseful sequences and reduced agency during
their execution, and version 4 contained both hints and the suspense sequences. I anticipated
that the sense of suspense in the version with both transformations will be rated higher in the
survey than all of the other three versions.
5.5 Results

64 people played the game and all of them completed the post game survey. 17 people each played version 1 (- hints/- suspense) and version 2 (+ hints/- suspense), and 15 people each played version 3 (- hints/+ suspense) and version 4 (+ hints/+ suspense). Using a series of one-tailed Mann-Whitney U tests, version 4 was compared to the other three versions. The other versions were compared among each other using two-tailed Mann-Whitney U tests. This was because no claims about one of these versions being rated higher than the others were made. Mann-Whitney U was used because it tests to see if a measurement from a population of ranked, non-parametric data is significantly different from another. The results of these tests are displayed in Table 5.1.

The first two player goals could be completed in either order. To see if the order of the completion affected player responses in Version 4, a two-tailed Mann-Whitney U test was performed comparing the responses for the first and second goals that the players performed. There was

<table>
<thead>
<tr>
<th>Version</th>
<th>Key</th>
<th>4v1</th>
<th>4v2</th>
<th>4v3</th>
<th>3v2</th>
<th>3v1</th>
<th>2v1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>0.006</td>
<td>0.01</td>
<td>0.003</td>
<td>0.66</td>
<td>1</td>
<td>0.69</td>
<td></td>
</tr>
<tr>
<td>Q2</td>
<td>0.34</td>
<td>0.06</td>
<td>0.03</td>
<td>0.87</td>
<td>0.12</td>
<td>0.19</td>
<td></td>
</tr>
<tr>
<td>Q3</td>
<td>0.09</td>
<td>0.17</td>
<td>0.07</td>
<td>0.58</td>
<td>0.94</td>
<td>0.66</td>
<td></td>
</tr>
<tr>
<td>Q4</td>
<td>0.08</td>
<td>0.21</td>
<td>0.54</td>
<td>0.35</td>
<td>0.12</td>
<td>0.47</td>
<td></td>
</tr>
<tr>
<td>Q5</td>
<td>0.05</td>
<td>1</td>
<td>0.03</td>
<td>(0.06)</td>
<td>0.88</td>
<td>(0.08)</td>
<td></td>
</tr>
<tr>
<td>Dynamite Q1</td>
<td>0.05</td>
<td>0.14</td>
<td>0.14</td>
<td>0.75</td>
<td>0.81</td>
<td>0.46</td>
<td></td>
</tr>
<tr>
<td>Q2</td>
<td>0.28</td>
<td>0.85</td>
<td>0.16</td>
<td>0.009</td>
<td>0.68</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>Q3</td>
<td>0.03</td>
<td>0.97</td>
<td>0.96</td>
<td>1</td>
<td>0.001</td>
<td>0.0007</td>
<td></td>
</tr>
<tr>
<td>Q4</td>
<td>0.23</td>
<td>0.49</td>
<td>0.23</td>
<td>0.48</td>
<td>0.95</td>
<td>0.48</td>
<td></td>
</tr>
<tr>
<td>Q5</td>
<td>0.44</td>
<td>0.96</td>
<td>0.95</td>
<td>1</td>
<td>(0.10)</td>
<td>(0.08)</td>
<td></td>
</tr>
<tr>
<td>Loot Q1</td>
<td>0.007</td>
<td>(0.09)</td>
<td>0.15</td>
<td>0.71</td>
<td>(0.08)</td>
<td>0.13</td>
<td></td>
</tr>
<tr>
<td>Q2</td>
<td>(0.10)</td>
<td>(0.06)</td>
<td>(0.06)</td>
<td>1</td>
<td>0.67</td>
<td>0.67</td>
<td></td>
</tr>
<tr>
<td>Q3</td>
<td>0.32</td>
<td>0.18</td>
<td>0.84</td>
<td>0.12</td>
<td>0.21</td>
<td>0.64</td>
<td></td>
</tr>
<tr>
<td>Q4</td>
<td>(0.07)</td>
<td>0.19</td>
<td>0.002</td>
<td>0.04</td>
<td>0.19</td>
<td>0.56</td>
<td></td>
</tr>
<tr>
<td>Q5</td>
<td>0.04</td>
<td>0.02</td>
<td>0.15</td>
<td>0.28</td>
<td>0.48</td>
<td>0.59</td>
<td></td>
</tr>
</tbody>
</table>
no significant difference in responses for survey questions 2-5, but players rated the second goal that they performed as significantly more suspenseful (p = 0.04) for question 1: “I felt excited about the outcome of the goal.”

5.6 Discussion

Figure 5.3 shows a visualization of the Mann-Whitney U tests for the key and loot goals. There are two ways to view the results: comparing responses across versions or across questions. When viewing across versions, Version 4 had the most improvement over the versions in this order: 1, 3, 2. This is interesting because it suggests that the presence of the suspense sequences (transformation 2) alone is not sufficient to evoke a high level of suspense without also providing game hints (transformation 1).

When viewing across questions, Version 4 had the most improvement over the other versions in questions 1, 2, and 5. This is encouraging because questions 1 and 2 both directly measure the player’s sense of emotion while playing the game, which the full treatment seemed to increase. The lower improvements in questions 3 and 4 may have been caused by player genre expectations. Question 3 was “I felt like an important story event was about to happen.” Since players were given these three goals explicitly, they may have felt that, across versions, completing these goals would advance the story. Question 4 was “I was uncertain about the successful completion of the goal.” Again, across versions, players may have had the genre expectation that if they are being asked to complete a goal, than there will be some way to complete it.

There was not a significant improvement in Version 4 with regards to suspense for the goal of obtaining the dynamite. This may be because the actions needed to complete the goal might not have been amenable to plan failure, the method IRIS uses to create suspense. The steps to get the dynamite may have seemed to contain less risk than the steps required to get the key or the loot, thus decreasing the overall sense of suspense when the player’s plan was thwarted. This suggests that in addition to applying the IRIS formula and the transformations needed to make the experience interactive, the goals that are targeted as suspenseful need to contain high risk actions that will concern the player when plan failure is introduced by the system.

5.7 Further Analysis

The second transformation, ensuring that players did not bypass the suspenseful story moments generated by IRIS, was realized in the game by a two-fold process: a forced encounter with
Figure 5.3 A visualization of the results of the one-tailed Mann-Whitney U tests comparing Key and Loot goals in Version 4 to those goals in the other three versions. A green box indicates that Version 4 was rated significantly higher with regards to suspense for that survey question, $p = 0.05$, yellow indicates marginal significance, $p = 0.10$, and black indicates no significant difference.
the suspenseful action sequence and a restriction of player actions during the execution of this suspenseful action sequence so that the player does not bypass the conflict. A follow-up experiment was performed to separate these two processes to see if they were both necessary to create suspense in the players. A fifth version of the game, where the players were forced to encounter the suspenseful action sequence but where their actions were not restricted, was created and evaluated.

In Version 4, the player could get the dynamite from the mine or the shopkeeper and could get the key from the teller or the teller’s room. Whichever method the player tried second allowed them to succeed. In Version 5, the player could get the item directly by killing the shopkeeper or teller, effectively bypassing the conflict that IRIS was trying to create. In Version 4, to get the loot out of town, the player was forced into a final conflict with the bandit. In Version 5, the player could kill the bandit during the story, avoid the bandit altogether, or talk to other characters during the final conflict with the bandit. All of these actions reduce the suspense of the conflict.

13 people played Version 5. Version 4 and 5 were compared using a two-tailed Chi-Squared test. The results are shown in Table 5.2. Versions 4 and 5 were not significantly different in terms of player suspense. This suggests that the restriction of player actions during the execution of suspenseful action sequences may not be essential to creating suspense in the player. However, the restriction of player actions even in Version 4 was kept to a minimum to ensure that the player did not feel railroaded into making a particular choice. The difference between the versions may have been more pronounced if there was a greater difference in actions available to the player between the two versions.

There was a non-statistically significant but possible improvement in Version 4 over Version 5 in the dynamite goal and in the questions "I felt anxious about the outcome of the goal" and "I felt that the outcome would be significant to future events in the story." This follow-up experiment indicates that restricting player actions during suspenseful action sequences may not be essential to creating suspense in the player but can still lead to some benefit if it is applied.

Version 5 was compared to Versions 1-3 using the one-tailed Mann-Whitney U test. The results of these tests are shown in Table 5.3. Version 5 did not perform significantly higher in terms of suspense when compared to Versions 1-3. This indicates that if the game designer is going to apply both of the transformations proposed for this experiment: a forced encounter with the suspenseful action sequence and a restriction of player actions during the execution of this suspenseful action sequence so that the player does not bypass the conflict, then player agency should be restricted as in Version 4, which did show an improvement over Versions 1-3.
Table 5.2 p values for the two-tailed Chi-Squared test comparison of Version 4 and Version 5 of the interactive text game. The numbers in parentheses indicate p values resulting from Barnard’s exact test where Chi-Squared could not be applied. There is not a significant difference between the two versions in terms of player suspense, but there is improvement in Version 4 over Version 5 in the dynamite goal, Question 2, and Question 5.

<table>
<thead>
<tr>
<th>Key</th>
<th>Question 1</th>
<th>Question 2</th>
<th>Question 3</th>
<th>Question 4</th>
<th>Question 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>0.39</td>
<td>0.08</td>
<td>0.96</td>
<td>0.72</td>
<td>(0.17)</td>
</tr>
<tr>
<td>Dynamite</td>
<td>0.13</td>
<td>0.12</td>
<td>0.73</td>
<td>(0.14)</td>
<td>0.14</td>
</tr>
<tr>
<td>Loot</td>
<td>0.50</td>
<td>0.21</td>
<td>0.93</td>
<td>0.30</td>
<td>0.07</td>
</tr>
</tbody>
</table>

5.8 Conclusion

In this experiment, I evaluated the IRIS system’s ability to create suspense in the context of an interactive text adventure game. It was hypothesized the translation into an interactive domain required two transformations from the original IRIS story output: 1) an in-game method of giving hints of how to achieve the game’s goals and 2) the presence of the suspenseful action sequences from the original story outline and the restriction of player agency during their execution. Participants played one of four versions of the game where one, both, or neither of the transformations were present. The results of this experiment show that for two of the three goals targeted to be suspenseful, the game version with both transformations was rated as significantly more suspenseful than the other versions. It was also shown that, while not essential to creating suspense, restricting player actions during the execution of the suspenseful action sequences may provide some benefit to suspense for the audience. This shows that the IRIS system is effective at creating suspense in an interactive text-based adventure game when both transformations are applied.
5.8. **CONCLUSION**

CHAPTER 5. **INTERACTIVE EVALUATION**

Table 5.3 The results of the one-tailed Mann-Whitney U tests comparing Version 5 to Versions 1-3 of the game for each of the three suspenseful goals. The values in bold indicate significant difference with regards to reported suspense for that survey question, \( p = 0.05 \), and the values in parentheses indicate marginal significance, \( p = 0.10 \).

<table>
<thead>
<tr>
<th>Key</th>
<th>5v1</th>
<th>5v2</th>
<th>5v3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Q1</strong></td>
<td><strong>0.03</strong></td>
<td>0.15</td>
<td><strong>(0.07)</strong></td>
</tr>
<tr>
<td><strong>Q2</strong></td>
<td>0.96</td>
<td>0.29</td>
<td>0.34</td>
</tr>
<tr>
<td><strong>Q3</strong></td>
<td>0.23</td>
<td>0.32</td>
<td>0.20</td>
</tr>
<tr>
<td><strong>Q4</strong></td>
<td>0.37</td>
<td>0.43</td>
<td>0.20</td>
</tr>
<tr>
<td><strong>Q5</strong></td>
<td>0.20</td>
<td>0.50</td>
<td>0.18</td>
</tr>
<tr>
<td><strong>Dynamite</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Q1</strong></td>
<td><strong>0.007</strong></td>
<td><strong>0.01</strong></td>
<td><strong>0.03</strong></td>
</tr>
<tr>
<td><strong>Q2</strong></td>
<td>0.47</td>
<td>0.94</td>
<td>0.39</td>
</tr>
<tr>
<td><strong>Q3</strong></td>
<td><strong>0.03</strong></td>
<td>0.24</td>
<td>0.25</td>
</tr>
<tr>
<td><strong>Q4</strong></td>
<td>0.37</td>
<td>0.18</td>
<td>0.50</td>
</tr>
<tr>
<td><strong>Q5</strong></td>
<td>0.20</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td><strong>Loot</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Q1</strong></td>
<td><strong>(0.10)</strong></td>
<td>0.42</td>
<td>0.50</td>
</tr>
<tr>
<td><strong>Q2</strong></td>
<td>0.50</td>
<td>0.44</td>
<td>0.46</td>
</tr>
<tr>
<td><strong>Q3</strong></td>
<td>0.42</td>
<td>0.32</td>
<td>0.37</td>
</tr>
<tr>
<td><strong>Q4</strong></td>
<td>0.41</td>
<td>0.24</td>
<td>0.17</td>
</tr>
<tr>
<td><strong>Q5</strong></td>
<td>0.37</td>
<td>0.35</td>
<td>0.25</td>
</tr>
</tbody>
</table>
Chapter 6

Conclusions

6.1 Implementation

The IRIS system creates suspenseful story outlines that can be implemented by an author in a variety of story media. The author provides domain information about the story world, a set of characters, and a set of goals around which the system will create suspense. IRIS uses a technique called planning to produce the story outline. Planning is the process by which an initial story state is translated into a desired goal state by a series of transformations called actions. At the center of the planning process is the protagonist character. The protagonist has a mental framework called Belief/Desire/Intention, by which its wants and goals are separated from the rest of the story world. This can allow for the protagonist to have beliefs at odds with the true state of the world, which is how the IRIS system creates suspense.

To create suspense around a given goal, the system selects an action to threaten that contributes to the goal. An antagonistic character forms a plan that will complicate, but not totally prevent, the completion of the suspenseful goal. This series of antagonistic actions is inserted into the planning process and the actions are made known to the audience but not the protagonist. This results in dramatic irony, where a disparity of knowledge between the audience and the protagonist results in anxiety and suspense in the audience because the protagonist is unknowingly working towards a doomed plan. The suspense is brought to a climax when the protagonist realizes its plan failure and must replan to try to complete the goal.

The result of the generation process is a sequence of story actions that contains localized moments of action-level suspense. The output is then given to an author, who can translate the story outline into his or her desired medium, such as text, video games, film, etc. The author adds the desired discourse into the story, which creates the author’s intended narrative effect.
6.2 Validation

The goal of my research was to create a planning-based narrative generation system that creates suspenseful story outlines. The system was demonstrated to be effective at creating suspense in both a non-interactive and an interactive narrative context. To demonstrate these claims, I addressed the following questions:

1. How can suspense be integrated into a narrative planning system?

2. Does the output of IRIS system create suspense in a reader in the context of a non-interactive text story?

3. Does the output of IRIS system create suspense in a player in the context of an interactive text game? What, if any, considerations can be made to adapt the IRIS story output to an interactive context?

To address the first question, “How can suspense be integrated into a narrative planning system,” I used the Artificial Intelligence technique of planning to generate a story outline. Within the planning process, I chose to create localized, story-action level moments of suspense around protagonist plan failure. I drew upon the Suspenser narrative generation system’s [8] definition of suspense: “the feeling of excitement or anxiety that audience members feel when they are waiting for something to happen and are uncertain about a significant outcome.” To model the protagonist’s mental state, I implemented a belief/desire/intention framework [9]. This allowed for the character to modify its plans and intentions when plan failure was introduced.

To address the second question, “Does the output of IRIS system create suspense in a reader in the context of a non-interactive text story,” I evaluated the IRIS system’s suspense generating component against several other suspense generation methods: a human author, an operationalization of the MINSTREL system [41], and a random, temporally consistent baseline. The different methods generated suspenseful story fragments which were evaluated by human subjects. It was found that the IRIS system was rated significantly higher in suspense than the
MINSTREL system and the random baseline and comparable to a human author composing fragments from the same domain.

To address the third question, “Does the output of IRIS system create suspense in a player in the context of an interactive text game? What, if any, considerations can be made to adapt the IRIS story output to an interactive context,” two transformations on the output of the IRIS system were proposed to adapt the output to an interactive context. The transformations were 1) an in-game method of giving hints of how to achieve the game’s goals and 2) the presence of the suspenseful action sequences from the original story outline and the restriction of player agency during their execution. Output from the IRIS system was used to create an interactive text adventure game. Four different versions of this game with the presence or absence of these transformations were played by human subjects. It was found that for two of the three targeted suspenseful goals, the version with both of theses transformations was rated as suspenseful and significantly more so than the three other versions. Also, while not essential to creating suspense, restricting player actions during the execution of the suspenseful action sequences may lead to a greater sense of suspense in the audience. This validates the claim that the IRIS system can create suspense in an interactive domain when both of the proposed transformations are applied.

The IRIS story generation system contributes to both the fields of non-interactive and interactive story creation. In non-interactive text stories, the author can feed in initial conditions, his or her story goals, and the protagonist’s initial intentions. The IRIS system will output a story outline that contains suspense around the protagonist to which the author can add discourse and other narrative properties. In an interactive narrative like a video game, a major time investment for the creative team is the creation of the narrative story events. The IRIS system can provide a template to the game designer so that they do not need to write all of the suspenseful moments by hand. The template nature of the IRIS output allows for flexibility in the way that the output is instantiated into the game, allowing choice of how the suspense is realized in the game.

6.3 Contributions

The IRIS system provides an operationalization of a narrative definition of suspense, “the feeling of excitement or anxiety that audience members feel when they are waiting for something to happen and are uncertain about a significant outcome” [8] and the means by which to create suspense in this fashion. This formal basis for generation of a narrative concept is valuable because it provides a way for scientists to form a common understanding of a creative phenomenon.
IRIS is also beneficial to authors because it can provide a suspenseful basis for a story while also allowing for the author’s personalization and choice of discourse details. IRIS’s ability to generate content for a variety of media allows for an alternative to the author to writing all of the story content by hand. The output’s template-based nature also allows for flexibility in the way suspense is realized in the author’s story.

6.4 Limitations

The IRIS system has made contributions to the field narrative generation by providing a framework for suspenseful story creation which can be instantiated in both non-interactive and interactive domains. However, there are some limitations of the system and also future work which can expand the system’s capabilities.

6.4.1 Limited Definition of Suspense

As mentioned in Section 2.5, there are many different kinds of suspense and no one unified definition of suspense. The IRIS system takes one particular definition from another successful narrative generation system and creates localized, story-action level moments of suspense. It does not try to capture all kinds of suspense that can be created in narratives, rather limiting itself to one specific kind of suspense that lends itself well to the application of planning for story generation. Because of the choice of definition, it does not capture all of the narrative phenomena that a reader may categorize as suspense. This could be addressed by attempting to synthesize a definition of suspense gathered from a variety of experts in narratology. The definition would have to be both validated and also operationalizable in a planning system.

6.4.2 Simple Suspense Templates

The suspense templates that IRIS uses to add suspense to the narrative generation process are fairly simple. They were created to be as general as possible for use in different domains, which is why there is no commitment in the templates themselves about how they are translated into story discourse. This results in the system output containing high level actions that require the user to both translate them from LISP-like syntax into human readable form and also add in discourse details. Also, the templates are all variations on the concept of dramatic irony, where an antagonistic force causes a conflict that is known to the audience but unknown to the protagonist. There are kinds of suspense other than dramatic irony that the system is not
capturing. This could be addressed by an expansion of the number or expressiveness of the suspense templates.

6.4.3 Limited Conflict Action Selection

The part of the algorithm where an action from the protagonist’s plan is selected to be interfered with by an antagonist is somewhat simplistic. The algorithm chooses an action that is labeled as suspenseful by the domain author. However, this action labeling not fine grained or context sensitive. Additionally, this is more work for the domain creation author and also offloads some of the expressiveness of the system from the story generation algorithm to the story domain. There exist some domains where an action being amenable to or not to suspense is not binary and also where actions may be suspenseful at certain times in the story or under certain conditions but not others. Instead of labeling the suspensefulness of actions, the characteristics of the action and its use in the story could be analyzed. For example, the system could look at the number of preconditions of the actions. If the action requires a large number of preconditions, then it might be harder or take longer to satisfy and thus might be a good target for an antagonistic character to thwart.

6.5 Future Work

While the generative power of IRIS allows for the creation of suspenseful story outlines, the system could be further expanded to allow for a more robust narrative structure and could also be integrated with other narrative generation systems. These additions would allow for the generation of more narratively rich stories.

6.5.1 Expand Character BDI Modeling and Protagonists Role

Since the IRIS system creates suspense around the protagonist of the story, a solitary protagonist is the only character with a robust mental model. However, the IRIS system could be expanded to provide mental modeling for supporting characters or for multiple protagonists. In this case, the system would need to decide which characters can contribute to the story plan and which characters solely work as support characters. The system would also need a way to integrate multiple protagonist plans into one unified plan. The extension of the character mental modeling may allow for increased believability of the actions of the other characters which can result in greater immersion and suspension of disbelief for the readers. This is because it
will look like multiple characters are acting to fulfill personal goals and not just performing actions in the service of a single protagonist’s plan. Additionally, multiple acting character could allow for a division of authorial goals. The system would then need to decide on how to divide the goals and how to reconcile different protagonists being at odds over the completion of an authorial goal.

6.5.2 Integration with Discourse Generation System

The IRIS system produces a suspenseful story outline without a commitment to how that outline is translated into discourse. The user of the system needs to translate the IRIS output into the desired medium and add his or her own discourse to the story actions. This is somewhat undesirable because it requires work on the user’s part and also because the effectiveness of the IRIS output is dependent on the user making an adding discourse. Instead of leaving this translation up to the user, the IRIS story output can be fed into a discourse generation system, like Suspenser [8]. This would produce an artifact that contains both the benefits of the story-action level suspense of the IRIS system and the contribution of the discourse generation system.
BIBLIOGRAPHY


APPENDICES
Appendix A

Evaluation of IRIS in a Non-Interactive Context

Figure A.1 shows the initial bank of IRIS suspense templates that were evaluated in the first experiment. A majority subset of them was found to be suspenseful, specifically the templates where the audience had knowledge about the protagonist’s imminent plan failure and the protagonist did not.

Figures A.2, A.3, and A.4 show the bank of sentences that the different story fragment generation processes had from which to select when creating story fragments for the initial experiment of IRIS in a non-interactive story context. The generation rules were the following: the story fragments were limited in length from 1 to 6 sentences, all sentences from phase 1 had to come before all sentences from phases 2 and 3, and all sentences from phase 2 had to come before all sentences from phase 3. This was to ensure a consistent temporal ordering of story events.

Figure A.5 shows the template based grammar in the follow-up experiment of IRIS in a non-interactive context. The grammar was designed to minimize the discourse level suspense that was inherent in the sentences from the initial experiment.

Figures A.6 and A.7 show information provided to the participants of the experiment: background story information and a screenshot of the survey.
APPENDIX A. EVALUATION OF IRIS IN A NON-INTERACTIVE CONTEXT

Figure A.1 List of the initial IRIS suspense templates. The templates with an asterisk were not rated as suspenseful in the evaluation.
The wizard tells the knight where the dragon is hiding.
The wizard tells the knight where to find the axe that will help him fight the dragon.
A stranger lies to the knight and tells him that the bridge the knight needs to cross is out. He
will not be able to get across as he intends.
The wizard tells the knight that he is not as properly equipped to fight the dragon as he
thought he was.
The knight sees the dragon fly to its lair.
The knight spots the axe that will help him fight the dragon.
The knight looks at the bridge and mistakenly thinks that the way across this bridge is out. He
will not be able to get across as he intends.
The knight is about to wade through the river but realizes that it is too deep.
The knight sees that he is not as properly equipped to fight the dragon as he thought he was.
The knight mistakenly thinks that he is not as properly equipped to fight the dragon as he
thought he was.
The knight tried to wade through the river but realized it was too deep.
The knight searches for a sword to kill the dragon and finds one.
The knight fought off the evil imps.
The knight healed the sick man with some herbs.
The knight saved a villager from a bear.
Unbeknownst to the knight, the knight’s armor will not adequately protect the knight from
the dragon’s attacks.
Someone filled up the last spot on a boat across the river. The knight sees that he will have to
find another way across.

Figure A.2 The sentence bank for phase 1 of the initial experiment of IRIS’s non-interactive story
generation
APPENDIX A. EVALUATION OF IRIS IN A NON-INTERACTIVE CONTEXT

The wizard tells the knight that his path to get to the dragon will not lead him there as he planned.
An orc lies to the knight and tells him that his sword will not hurt the dragon.
The knight spots the axe.
The knight looks at his sword and mistakenly thinks that it is too dull to kill the dragon, as he intends.
The knight sees that his path to get to the dragon will not lead him there as he planned.
The knight fails to acquire the axe.
The knight fails to reach the dragon’s location.
An orc steals one of the knight’s weapons that the knight had planned to use to fight the dragon.
The knight finds where the dragon is hiding.
The knight became trapped in a rock slide. He tried to get out of the way but failed.
A troll destroys a magic gem that could have been used to kill the dragon.
An archer who could have helped kill the dragon leaves town.
Unbeknownst to the knight, the dragon knows the knight is coming.
(Elsewhere, Unbeknownst to the knight), - the axe is stolen by a kobold.
(Elsewhere, Unbeknownst to the knight), - The dragon flies to a new location.

Figure A.3 The sentence bank for phase 2 of the initial experiment of IRIS’s non-interactive story generation

The knight sees the dragon.
The dragon begins to fly away to safety out of range of the knight.
The dragon attacked the knight.
The knight tried to fight the dragon but fails.
The knight attacks the dragon but fails to kill it.
The dragon attacks the knight. The knight tried to retreat from the dragon but failed.
The knight swung at the dragon but missed. He saw that it was because the dragon was anticipating his movements.
The knight’s armor is destroyed by the dragon’s attack.

Figure A.4 The sentence bank for phase 3 of the initial experiment of IRIS’s non-interactive story generation
APPENDIX A. EVALUATION OF IRIS IN A NON-INTERACTIVE CONTEXT

Actions
<person> observes <condition>.
<person1> conveys to <person2> <condition>.
<person1> attacks <person2>.
<person1> forces <person2> to go from <place1> to <place2>.
<person> moves to <place>.
The detective wants to question <person>.
The detective tries to question <person>.
The detective wants to arrest <person>.
The detective tries to arrest <person>.

Conditions
that <person> is dead.
that <person> is at <place>.
that <person> is not at <place>.

Modifiers
(... but fails)
(Unbeknownst to the detective, Without the detective’s knowledge, Unknown to the detective ...)
(Elsewhere, In another location, Somewhere else ...)

People
The maid
The butler
The chef
The house guest
The detective

Places
The dining room
The attic
The basement
The living room

Figure A.5 The template-based grammar for story fragment creation in the follow-up experiment of IRIS’s non-interactive story generation
“This story takes place as a murder mystery. There is a detective, who is the protagonist, a maid, a butler, a chef, and a houseguest. The detective’s goal is to capture who killed the host. To do this, the detective needs to:
1 Gain knowledge of who killed the host. This is done by questioning the suspects, which will cause them to reveal what they know,
2 Be in the same location as the killer.
3 Arrest the killer, which requires the detective to suspect the person he is arresting is the killer.

The following story fragments are trying to create suspense around the detective’s plan to achieve this goal.

Suspense is defined as ‘the feeling of excitement or anxiety that audience members feel when they are waiting for something to happen and are uncertain about a significant outcome.’

Figure A.6 The background information given to the participants about the story world and the desire to create suspense.
APPENDIX A. EVALUATION OF IRIS IN A NON-INTERACTIVE CONTEXT

This story takes place as a murder mystery. There is a detective, who is the protagonist, a maid, a butler, a chef, and a houseguest. The detective’s goal is to capture who killed the host. To do this, the detective needs to:

1) Gain knowledge of who killed the host. This is done by questioning the suspects, which will cause them to reveal what they know.
2) Be in the same location as the killer.
3) Arrest the killer, which requires the detective to suspect the person he is arresting is the killer.

The following story fragments are trying to create suspense around the detective’s plan to achieve this goal.

Suspense is defined as 'the feeling of excitement or anxiety that audience members feel when they are waiting for something to happen and are uncertain about a significant outcome.'

Please note, it is extremely important that you do not use the browser’s back button or refresh the page. Nothing bad will happen, but you will likely encounter an error and have to begin again.

**Comparison 1/36**

<table>
<thead>
<tr>
<th>Story Fragment A</th>
<th>Story Fragment B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fragment A:</td>
<td>Fragment B:</td>
</tr>
<tr>
<td>The detective tries to question The houseguest.</td>
<td>The detective observes that The chef is at The attic.</td>
</tr>
<tr>
<td>The detective wants to arrest The chef.</td>
<td>The detective wants to question The chef.</td>
</tr>
<tr>
<td>The detective moves to The dining room.</td>
<td>The maid forces The chef to go from The attic to The dining room.</td>
</tr>
<tr>
<td>The detective observes that The chef is at The dining room.</td>
<td>The detective moves to The attic.</td>
</tr>
<tr>
<td>The detective tries to arrest The chef.</td>
<td>The detective tries to question The chef but fails.</td>
</tr>
<tr>
<td>Story Fragment A Suspenseful and Story Fragment B Was Not</td>
<td>The detective observes that The chef is not at The attic.</td>
</tr>
<tr>
<td>Story Fragment B Suspenseful and Story Fragment A Was Not</td>
<td></td>
</tr>
<tr>
<td>Both Were Suspenseful</td>
<td></td>
</tr>
<tr>
<td>Neither Were Suspenseful</td>
<td></td>
</tr>
</tbody>
</table>

[continue]

Figure A.7 A screenshot of the survey, where participants read and compared the suspensefulness of two story fragments.
Appendix B

Evaluation of IRIS in a Interactive Context

Figure B.1 shows the survey questions presented to the participants in the evaluation of IRIS in an interactive domain. The participants were given a five point Likert scale with which to answer the questions. These questions were generated as an operationalization of IRIS’s definition of suspense. Figure B.2 shows a screenshot of an example question from the survey.

Question 1: I felt excited about the outcome of the goal
Question 2: I felt anxious about the outcome of the goal
Question 3: I felt like an important story event was about to happen
Question 4: I was uncertain about the successful completion of the goal
Question 5: I felt that the outcome would be significant to future events in the story

Figure B.1 The survey questions presented to the participants in the post game survey. These questions were generated by an operationalization of IRIS’s definition of suspense.
Figure B.2 A screenshot showing an example question from the post game survey presented to the participants.
Appendix C

The Western Bank Heist Domain

Figures C.1, C.2, C.3 enumerate the actions available in the Western heist domain. They are used in the sample story generation example from Section 3.6 and the experimental evaluation of the IRIS system in the domain of an interactive text adventure game. Figure C.4 shows an example of the start state of one of the four versions of the text adventure game.
Appendix C. The Western Bank Heist Domain

Figure C.1 Part 1 of the actions available in the Western domain, used in the example in Section 3.6 and the evaluation of IRIS in an interactive domain.
APPENDIX C. THE WESTERN BANK HEIST DOMAIN

Figure C.2 Part 2 of the actions available in the Western domain, used in the example in Section 3.6 and the evaluation of IRIS in an interactive domain.
Figure C.3 Part 3 of the actions available in the Western domain, used in the example in Section 3.6 and the evaluation of IRIS in an interactive domain.
APPENDIX C. THE WESTERN BANK HEIST DOMAIN

Figure C.4 Example initial state of the world for the Western heist domain. This was the +hints/+suspense version of the game.