ABSTRACT

AMOS-BINKS, ADAM ALEXIS. Intention Revision of Plan-Based Agents for Narrative Generation. (Under the direction of R. Michael Young and David Roberts).

Narrative is context where intention plays a critical role; individual characters collaborate and conflict while in pursuit of their intentions. We often experience suspense when a character’s intention is in doubt, or relief when a character’s malicious intent is foiled. This range of cognitive responses we exhibit to narrative leads some to characterize narrative as a fundamental cognitive ability of humans. Interactive narrative build on these foundations and is a powerful platform that engages our cognitive abilities through computational models of narrative phenomena. Its applications are wide ranging from education, to entertainment, and even to business (e.g. strategic foresight).

One of the more ambitious goals of interactive narrative is for an experience manager, an agent who ensures certain experience qualities, to adapt the story structure to accommodate user actions that introduce inconsistencies to a story’s plot. Adapting the story structure in this way, which I refer to as a story branch, creates a virtual environment where a user is free to exact their agency. However, enabling agency by accommodating user actions is at odds with intentional plans that use the IPOCL representation. User actions can introduce causal link threats to the planned steps of a non-player character agent, making an agent’s goal impossible to achieve.

To address this limitation of an experience manager, I make three major contributions in this dissertation. First, I formalize intention revision for intentional agents in a way that is consistent with human comprehension. I apply existing theories of belief-desire-intention agents and intentional logic to formalize definitions of intention revision for intentional planning. My evaluation shows that after experiencing the intention revision model I formalized, subject’s have an accurate mental model of intention revision.

Second, I develop an algorithm that characterizes a planning problem’s solution space in a way that supports the formal definition of intention revision. The algorithm captures sets of interdependent agent goals that represent the causally-necessary agent goals to reach a given state. Results on benchmark problems show that when a planner uses the algorithm to direct planning, the planner achieves a more diverse plan-set that is a representative characterization of the solution space.

Lastly, I investigate whether narrative texts that contain intention revisions generated from the aforementioned algorithm lead to greater subject engagement. Using intentional plans that contain intention revisions, I generate simple template-based natural language text that human subjects read. My analysis shows that subjects who read narrative texts where characters revise their intentions experience greater transportation and perceived interest than subjects who read texts where characters achieve their original intentions without issue.
© Copyright 2018 by Adam Alexis Amos-Binks

All Rights Reserved
Intention Revision of Plan-Based Agents for Narrative Generation

by
Adam Alexis Amos-Binks

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
2018

APPROVED BY:

---
R. Michael Young
Co-chair of Advisory Committee

David Roberts
Co-chair of Advisory Committee

James Lester
Christopher Martens
DEDICATION

To my family; my harvest and bond.
BIOGRAPHY

ADAM AMOS-BINKS was privileged to have led the Anticipatory Thinking research program at the Laboratory for Analytic Sciences while doing his dissertation. They eventually became one and the same pushing the program and his research in ways that would not have been possible otherwise. Previously he was a steaming machine learning researcher, where he worked in high performance computing for defending the nation’s communication networks.

In 2013, Adam was invited to become a visiting research scholar at the Laboratory for Analytic Sciences, a new government partnership with North Carolina State University and industry partners. In his other lives, Adam worked on Canada’s first dial-up internet service and joined a fantastic, but flawed wireless protocol startup.
ACKNOWLEDGEMENTS

Completing a dissertation is a herculean effort. I’ve had so much help along the way, and the following reverse-chronologically ordered list is just an attempt to give thanks. Riva Soucie, my wife and best friend, you were really there every step of the way for me. Michael Young, my advisor, I am grateful for your time and tutelage in the privilege of learning scientific research and I admire your pursuit of it in such an interesting field. Dave Roberts, my co-advisor, your perspective on what a PhD really is and breadth of knowledge on evaluations was always there when I needed it. Michael and Dave thank you both for supporting my ideas even when they were a full ten on the crazy idea scale. Mike Bender and Alyson Wilson from the Laboratory for Analytic Sciences (LAS). Your support of me developing the Anticipatory Thinking research program around my dissertation interests pushed my research in ways I can’t describe. D’Arcy Walsh and Renaud Levesque from CSE. Your belief in me and the vision of the LAS made all this possible. Dave Harris, the inaugural director of the LAS. Giving me the opportunity to be involved with the LAS from the first days is a great honour I will be forever grateful for. Joe McCloskey and Randy Dahlberg, your support for my integration into Math Research is the start of this dissertation’s causal chain.

A huge thanks to everyone else I didn’t name in the Liquid Narrative research group, the CIIGAR research group, the Center for Educational Informatics, NC State, the LAS, the intelligence community, as well as family, friends, and colleagues around the world.
TABLE OF CONTENTS

LIST OF TABLES ........................................................................ vii

LIST OF FIGURES ..................................................................... viii

Chapter 1 Introduction ........................................................... 1
  1.1 Motivation .................................................................. 2
  1.2 Problem .................................................................... 3
  1.3 Approach ................................................................... 4
  1.4 A Reader’s Guide ........................................................ 6

Chapter 2 Previous work ....................................................... 8
  2.1 Story Generation ......................................................... 8
  2.2 Intention Revision ....................................................... 9
  2.3 Narrative Comprehension .......................................... 11
  2.4 Problem Formulation ................................................ 13
  2.5 Solution Diversity ..................................................... 15
  2.6 Experience Management .......................................... 17
  2.7 Narrative Experience ............................................... 19

Chapter 3 Plan-Based Intention Revision ............................. 21
  3.1 Intentional Planning .................................................. 24
  3.2 Intention Revision ..................................................... 28
  3.3 Intention Revision Comprehension ............................. 32
  3.4 Evaluation Method ................................................... 35
    3.4.1 Subjects .......................................................... 36
    3.4.2 Materials ........................................................ 37
    3.4.3 Procedures ....................................................... 37
    3.4.4 Results ............................................................ 42
  3.5 Discussion ............................................................... 45

Chapter 4 Intention Dependencies for Intention Revision .... 50
  4.1 Intention Dependency Graph ..................................... 52
  4.2 Generating a Diverse Plan Set .................................... 63
    4.2.1 Plans from IDG Solution Vertices ....................... 63
    4.2.2 IDG Advantages and Limitations ....................... 64
  4.3 Evaluation ............................................................... 64
    4.3.1 Time and Space Complexity .............................. 65
    4.3.2 Designing a Problem with a Diverse Solution Space 67
    4.3.3 Efficiently Generating a Diverse Plan Set: Intra Plan Set Diversity 70
    4.3.4 Characterizing a Problem’s Solution Space: Inter Plan Set Diversity 72
  4.4 Discussion ............................................................... 74

Chapter 5 The Subjective Experience of Intention Revision .... 75
# Table of Contents

5.1 Intention Revision as a Branching Mechanism ........................................ 77  
5.2 Generating Story Branches ................................................................. 81  
5.3 Evaluation ......................................................................................... 84  
  5.3.1 Method ....................................................................................... 86  
  5.3.2 Results ....................................................................................... 94  
5.4 Discussion ......................................................................................... 96  

Chapter 6 Conclusion .................................................................................. 99  
  6.1 Contributions .................................................................................... 100  
    6.1.1 Intention Revision .................................................................... 100  
    6.1.2 Intention Dependencies ............................................................ 101  
    6.1.3 Subject Experience of Intention Revision ................................ 102  
  6.2 Future Work ..................................................................................... 103  
    6.2.1 Computational Model of Betrayal ............................................ 104  
    6.2.2 Procedural Semantics for Domain Authors ............................. 105  
    6.2.3 Subject Experience of Intention Revision ............................... 106  

BIBLIOGRAPHY ........................................................................................... 108
# LIST OF TABLES

| Table 2.1 | QUEST questions and legal answers | 11 |
| Table 2.2 | QUEST node descriptions | 12 |
| Table 2.3 | QUEST arc descriptions | 12 |
| Table 2.4 | Drama Manager desiderata descriptions and plan-based systems [RI08]) | 18 |
| Table 3.1 | Questions to assess intention revision explainability | 34 |
| Table 3.2 | Questions to evaluate the QKS intention revision subgraph | 35 |
| Table 3.3 | Narrative text design elements used to create experiment materials | 38 |
| Table 3.4 | Prison Sentence text | 38 |
| Table 3.5 | Farm Boy text | 39 |
| Table 3.6 | Contingency results for Prison Sentence questions | 46 |
| Table 3.7 | Contingency results for Farm Boy questions | 46 |
| Table 3.8 | QKS intention revision subgraph question results for Prison Sentence | 47 |
| Table 3.9 | QKS intention revision subgraph question results for Farm Boy | 48 |
| Table 4.1 | Summary of intentions necessary to solve the politico problem | 70 |
| Table 4.2 | Plan set diversity of story benchmarks [War14] compared against previous results [FW16]. | 71 |
| Table 4.3 | IDG as a characterization of solution space diversity | 74 |
| Table 5.1 | Transportation Scale (TS) items and subscales, (R) indicates reverse coding. | 89 |
| Table 5.2 | Perceived Interest Questionnaire (PIQ) items | 89 |
| Table 5.3 | Jail text generated from plans with no intention revision. | 90 |
| Table 5.4 | Jail text generated from plans with intention revision. | 91 |
| Table 5.5 | Wilderness text generated from plans with no intention revision. | 92 |
| Table 5.6 | Wilderness text generated from plans with intention revision. | 93 |
| Table 5.7 | Story branch summaries of text read by experimental groups | 94 |
| Table 5.8 | The table below contains the mean Transportation Scale (TS) and Perceived Interest Questionnaire (PIQ) scores, with variance in parentheses, for the control (C) and treatment (T) experimental groups of each Jail and Wilderness story variant. | 94 |
LIST OF FIGURES

Figure 2.1 Planning graph developed by Blum and Furst [BF97]. The alternating proposition and action layers eventually reach the goal propositions in the final layer from which a solution can be extracted. The solid lines indicate the precondition/effect relationships between propositions and actions. The move L-P action in time 1 requires the preconditions at RL and fuel R be in time 1. The dotted lines represent mutually exclusive relationships with line between move L-P and the fuel R proposition at time 2 indicating it is not possible for fuel R to be true after executing move L-P. .................................................. 14

Figure 2.2 The planning problem components (domain, initial state, goal state, key/objects) with corresponding problem formulation structures developed as part of the Glaive planner [War14]. The goal graph captures an agent’s subplans to achieve their intention and are used as a filter when adding actions to the plan graph. .......................................................... 16

Figure 3.1 The Prison planning problem Φ and its elements ................................................. 25
Figure 3.2 Prison intentional plan (π) including intention frames, I(π) ................................. 26
Figure 3.3 While executing π, a story branch (π′) is created in response to an exceptional action S′4 ................................................................. 29
Figure 3.4 Execution state of π after executing each step, s ∈ S(π) ........................................ 30
Figure 3.5 The QUEST knowledge structure intention revision subgraph for plan-based intention revision .................................................................... 34
Figure 3.6 The QUEST knowledge structure intention revision subgraph of the Prison example. 35
Figure 3.7 Experimental groups for intention revision experiments; note the differences in the story statements in each group ........................................... 36
Figure 3.8 QUEST Knowledge Structure (QKS) for Prison Sentence ................................... 40
Figure 3.9 Local goal hierarchy change and QKS intention revision subgraph for Prison Sentence .................................................................................. 41
Figure 3.10 QUEST Knowledge Structure (QKS) for Farm Boy ................................. 42
Figure 3.11 Local goal hierarchy change and QKS intention revision subgraph for Farm Boy 43

Figure 4.1 Original Prison plan π with an additional story branch option, π″, in response to DenyTrial .......................................................... 53
Figure 4.2 Goal graphs for Prison planning problem .............................................................. 54
Figure 4.3 SPG for Prison planning problem ........................................................................ 55
Figure 4.4 IDG for Prison planning problem ........................................................................ 56
Figure 4.5 Original Prison planning problem modified into IDG partition problems ......... 57
Figure 4.6 An example of the worst-case space IDG from 4 fictitious labels, dotted edges indicate the destination has been omitted for readability ........... 67
Figure 4.7 Politico problem ................................................................................................. 68
Figure 4.8 Politico solution plans for π1, π4, π9 ................................................................. 69
Figure 4.9 New solution plans found over time for the Politico problem ......................... 73

Figure 5.1 A taxonomy of accommodation types to causal-link threats ............................ 77
Figure 5.2  A modified Prison planning problem Φ that demonstrates the original plan (top) relative to the structure of a duplicate achievement story branch (bottom), DUP(π, χ, π′) ................................. 79
Figure 5.3  A modified Jail planning problem Φ that demonstrates the original plan (top) relative to the structure of a subplan story branch (bottom), SUBPLAN(π, χ, π′) 80
Figure 5.4  A modified Prison planning problem Φ that demonstrates the original plan (top) relative to the structure of a subgoal story branch (bottom), SUBGOAL(π, χ, π′) 81
Figure 5.5  A modified Jail planning problem Φ that demonstrates the original plan (top) relative to the structure of an intention revision story branch (bottom), IR(π, χ, π′). Note this is a duplicate figure first introduced in Section 3.2 as Figures 3.2 and 3.3 82
Figure 5.6  The Prison planning problem Φ and its elements. This is the same figure as Figure 3.1, re-introduced here for readability ................................. 83
Figure 5.7  Intention revision story structure ................................................. 87
Figure 5.8  Text for πT J 1 . The underlined text denotes the exceptional action that initiates an intention revision. ................................................. 88
Figure 5.9  Mean narrative transportation and perceived interest scores ................. 98
Figure 6.1  Hypothesized QKS subgraph of betrayal. .................................... 104
Figure 6.2  Procedural semantics view of the IDG from Figure 4.4. ................. 105
Figure 6.3  Interactive narrative experimental groups, an extension to the experiment represented in Figure 5.7. ...................................................... 107
CHAPTER

1

INTRODUCTION

From our criminal justice system to military planning, the concept of *intention* is pervasive in our lives. Criminals are sentenced to jail time when a jury is convinced there was *criminal intent* in their actions, irrespective of whether the accused succeeded in committing a crime or not. Military leaders provide desired end states and outcomes of military operations through *commander’s intent*, leaving tactical decisions on collaboration and co-ordination to subordinates in their hierarchy. Our reliance on intention’s ability to affect future behavior and action in such a profound way underscores how powerful the concept is in human cognition.

Narrative is another context where intention plays a critical role; individual characters collaborate and conflict while in pursuit of their intentions. We often experience suspense when a character’s intention is in doubt, or relief when a character’s malicious intent is foiled. This range of cognitive responses we exhibit to narrative leads some [Her03] to characterize narrative as a fundamental cognitive ability of humans. Often referred to as *narrative intelligence* [MS99], education [e.g. Row11] and entertainment [e.g. Gam12] leverage our narrative intelligence as a kind of cognitive shortcut to improve learning and artistic experiences.

Since the first early concepts of artificial intelligence, researchers thought it possible to represent a narrative as a plan [SA77], where a plan is a causally connected action sequence that achieves a planning problem’s end state. At the time, the formalization of classical planning was in its infancy and generating plans was only done at a small scale. From the fledgling formalization, classical planning has experienced several advances in representation [e.g. McD98], performance [e.g. Hel06], and applications [e.g. Mye99].
More recently, there has been a more concerted effort to extend planning representation to capture a
greater number of narrative phenomena [FY17; Por13; Por10; TY17; ABY18; Che15].

Of greatest interest to this work is the intentional, partial-order, causal-link (IPOCL) plan representa-
tion [RY10]. IPOCL representation is built on classical planning, but with the additional requirement
that character agents execute causally linked action sequences to accomplish their goals, in addition to
those in the planning problem. This additional requirement has operationalized the concept of intention
in plans that is consistent with a model of human comprehension of narrative [RY10]. Also notable is
an extension to IPOCL representation, known as the conflict, partial-order, causal-link (CPOCL) plan
representation [War14]. CPOCL represents conflict between two character agents who have conflicting
intentions by including failed attempts to reach their goals. IPOCL and CPOCL provide representations
of intentional agents that adopt and drop goals that are consistent with human perceptions of inten-
tion [RY10; War14; CR16]. Together these representations make up what I will refer to as intentional
planning.

First efforts of intentional planning focused on generating plans that when transformed into narrative
texts, humans would understand and consider cohesive. In addition to generating linear texts from
intentional plans and other representations, there was a growing community whose goal was to advance
the state of the art in interactive narrative [RB12]. In contrast to a linear narrative text, a basic interactive
narrative text periodically presents a user with choices, or story branches, that affect the structure of the
story. A user can then experience a wide variety of outcomes and story structures by pursuing different
branches of the story. Choose-your-own-adventure novels [e.g. Pac82] are a first example of interactive
narrative, with more modern versions implemented as computer games such as Mass Effect [Mic07] and
BioShock [2K 08]. The appeal of interactive narrative and branching stories is now widespread.

1.1 Motivation

The appeal of interactive narrative and branching stories is thought to be related to a user’s experience
of agency and locus of control when given the ability to make choices [Mat97; Mur98; RI08]. These
choices draw the user further into the story by forcing them to consider the first, second, and nth order
consequences of their choices, increasing their involvement and leading to more engaged experiences.
Underlying the choices presented to the user is a branching story structure. In choose-your-own-adventure
novels it is more obvious that the story branches are fixed, but even in modern computer games the story
branches are hard coded ahead of time by the authors.

To further increase a user’s overall experience through greater agency, the interactive narrative
research community seeks to move beyond predefined story branches and choice points to an environment
where a user is free to interact with the world in the manner they choose. This open world environment
requires the underlying narrative to adapt and change to fit the world state a user has created [You07]. To
this end, procedurally generated content and other generative representations (e.g intentional planning)
are the most prevalent mechanisms by which story branches are generated.

The deliberate and predefined story branches are individually crafted by authors so they implicitly preserve a degree of coherence and experience quality. In contrast, generating a story branch requires that coherence and experience quality be managed explicitly, as there is no human able to hand craft each story branch. They would be very quickly overwhelmed by story branch requests from an open world environment. Consequently, the interactive narrative research community has identified the need for an experience manager [Bat92; Rob07; Mag07; RI08; Rob11; Rob14] agent to ensure that a user’s experience maintains some measure of quality and authorial goals are maintained.

Of most interest to this dissertation is that at a very high level an experience manager agent is responsible for preserving the properties of the chosen story representation when creating a story branch. When intentional planning is the story representation, an experience manager is responsible for balancing the experience quality with the intentions and goal-oriented behavior of the character agents in the story branch. Bestowing an experience manager with this capability could be accomplished with a robust story branching mechanism consistent with human understanding of intention, along with the technical means to execute it.

1.2 Problem

The central problem to providing experience managers with robust branching mechanisms for intentional plans is the power of intention itself. Specifically, intentions of the character agents establish expectations of future behavior in users. When an experience manager creates a new story branch, careful attention needs to be paid to the user’s mental model and expectation of character behavior.

For example, a story branch where all the characters’ intentions suddenly change from the original story could be very jarring and incoherent when not connected to a reason or cause. Their actions in the story branch would be inconsistent with their established intentions, violating a user’s expectations of their behavior. In contrast, if an event occurs that provides compelling reasons and explains why characters changed their intentions then users may comprehend. While there is substantial complexity entangled in revising intentions for story branches, I distill it down into three sub-problems for intentional planning.

The first sub-problem is formalizing intention revision for intentional plans. Even from the first concepts of intention, discussions surrounding when and how agents go about revising their intentions began. While some formalizations exist in formal logic, no operationalization for intentional planning exists. Defining exactly when an agent would reconsider their intentions and how the deliberation steps consider their options are the principal components of a definition for intentional planning. Furthermore, due to the human-centric application context, it is desirable to understand exactly how users’ mental models of character behavior respond to formalized definitions of intention revisions in a story branch. Understanding the relationship between a formalized definition of intention revision and human mental
models gives an experience manager a characterization of a story branch’s effect on experience quality.

Even with a formalized definition of intention revision and a robust model of human understanding of it, there are still technical hurdles to overcome in plan generation. **A second sub-problem of intention revision is identifying a space of story branches.** Intention revision is being used to ensure that an interactive narrative experience continues, despite some complications introduced by a user. Therefore, searching for a story branch will include options that are not quite perfect and exact matches. With the prospect of ending the experience due the lack of a perfect story branch, a story branch that aligns most closely with human models of narrative comprehension will do. A characterization of the available story branches enables an experience manager to identify a story branch that has the most desirous effect on experience quality.

At a more fundamental level than the technical challenges of identifying story branches in a planning problem and the formal definitions of intention revision, there is a long-held assumption that readers, players, and users enjoy story branches. **A third and final sub-problem is to test the assumption of story branch enjoyment** as it applies to story branches created when using intention revision as a branching mechanism. An experience manager is responsible for ensuring that some properties of experience quality are preserved while a user experiences an interactive narrative. While this work is in support of an experience manager that ensures experience quality, intention revision needs to be assessed at a basic story level without interfering effects from interactive environments to evaluate intention revision’s effectiveness. Understanding intention revision’s effect on narrative experience alone will indicate its appropriateness as a branching mechanism for interactive narratives.

In the problems above, I have described challenges central to character agents adapting and changing their behavior so an experience manager can ultimately create more engaging interactive narrative experiences.

### 1.3 Approach

My approach to the overall problem of using intention revision to create story branches is captured by the following thesis:

“Using a plan-based model of intention revision that is consistent with human comprehension to generate a story branch will lead to greater subject engagement in generated narrative texts.”

The proposed thesis addresses the aforementioned sub-problems by answering the following research questions:

**RQ1** How can a plan-based formalization of intention revision be consistent with models of human comprehension of narrative?

**RQ2** How can a problem formulation methodology use the structure of agent intentions to efficiently characterize the solution space of an intentional planning problem?
RQ3 What factors affect the relationship between an intention revision and a subject’s engagement in narrative text?

To answer RQ1, I apply existing theories of belief-desire-intention agents and intentional logic to formalize definitions of intention revision for intentional planning. The formalization comprises two parts, the first being when an agent should reconsider their intentions. This has been a discussion point for some time [Bra03; CL90; Alc85; Van07]; for this context, however, reconsidering intentions is done when the interactive narrative would otherwise end due to some exceptional user action. Scoping reconsidering intentions to this case does not limit it to further application; rather it ensures it has more than just aesthetic properties. The second part of my formalization of intention revision is the deliberation steps an agent takes once they are reconsidering an intention. The first step an agent takes is considering whether to drop an existing intention. To do this, the agent must believe the intention is unachievable. If the intention is unachievable, the agent’s next step is to identify an intention that has a subgoal of the original intention. Should one exist, the agent will adopt it. Lastly, should no subgoal intention exist, an agent selects a new goal from a space of possible intentions. Together, these three steps capture the deliberation an agent executes when revising their intentions. I ensure the reconsideration and deliberation parts of intention revision are consistent with human comprehension by using the QUEST cognitive model of narrative comprehension. QUEST uses a knowledge structure (QKS) to represent a mental model of a story’s intentional structure. The QKS is a directed graph composed of nodes representing events, states, and goals with arcs that describe seven types of relationships between nodes. By traversing the QKS graph for question-answer appropriateness, we can glean insights into the human comprehension of stories. I define a QUEST knowledge structure subgraph of intention revision. Using this subgraph, I use QUEST’s question-answer protocol to derive three question-answer pairs that form the basis of a human subject evaluation [ABY18]. The evaluation shows that the QKS subgraph represents a user’s mental model after experiencing the intention revision model I formalized.

In contrast to my approach to RQ1, I take a more algorithmic-centric approach to answering RQ2. For an agent to execute the intention revision deliberation steps, a characterization of a planning problem’s solution space is required. To characterize a planning problem’s solution space of intentional plans, I exploit a fundamental property of intentional planning. Intentional planning requires that every step of a solution plan be part of an agent’s subplan to achieve their intention. Consequently, the steps in agent subplans will be dependent on one another. Through a structure I define, called the Intention Dependency Graph (IDG), I capture sets of interdependent agent goals that represent the causally-necessary agent goals to reach a given state. When used in conjunction with a planning graph, the IDG can characterize the solution space of a planning problem. Because the characterization is based on interdependent character goals, it also represents the intention revisions that would take place between an original plan and a story branch. I demonstrate the IDG’s ability to characterize a solution space by comparing it against existing diverse planners on narrative benchmark problems. The results show that when a planner uses
the IDG to direct the planning, it achieves a more diverse plan-set that is a representative characterization of the solution space. These results are significant as they demonstrate the value of a problem formulation method for diverse planning that analyzes the problem structure for promising and qualitatively different areas of the solution space.

Lastly, to answer RQ3 I integrate the results from the approaches used to address RQ1 and RQ2. Ultimately, creating a story branch should at worst present an attractive trade-off with the agency a user experiences. At best, story branches would improve overall experience. I investigate the effect intention revision has on reader’s experience of narrative text by simulating an interactive environment. I develop a planning problem where simulated user actions initiate a non-player character agent to reconsider their intention. As they follow the deliberation steps defined in the approach to RQ1, the solution space characterization supplied by the IDG from RQ2 is used. Using this planning domain along with the results from RQ1 and RQ2 in a simulated interactive narrative environment has two major benefits. First, it separates the effects of an interactive environment from those of intention revision. Using intentional plans that contain intention revisions, I generate simple template-based natural language text that human subjects read. A second benefit is that widely used and well validated measures of subject experience of narrative text can be used directly [e.g. GB00; Sch97]. After reading a narrative text, subjects complete the narrative transportation and perceived interest questionnaire. My analysis shows that subjects who read narrative texts where characters revise their intentions experience greater transportation and perceived interest than subjects who read texts where characters achieve their original intentions without issue.

In short, I make three major contributions in this dissertation that answer three research questions in support of the overall thesis. First, I formalize intention revision for intentional agents in a way that is consistent with human comprehension. Second, I develop an algorithm that characterizes a planning problem’s solution space in a way that supports the formal definition of intention revision. Lastly, I investigate whether narrative texts that contain intention revisions generated from the aforementioned algorithm lead to greater subject engagement.

1.4 A Reader’s Guide

Here in Chapter 1, I provided a very high-level summary of this dissertation without much background or details of previous work. In Chapter 2, I address this by reviewing the pertinent details of previous work and provide some insights on where it fits into the overall approach.

Some readers will be delighted to find a running example throughout the content chapters of this dissertation. It is very simple in its basic form, and I add to it as needed to illustrate and highlight more complex contributions. In Chapter 3, I use the most simple form of my running example to demonstrate the formalization of intention revision and how it translates to a QUEST knowledge structure and evaluation. Chapter 4 adds to the example to demonstrate the algorithms used to characterize the story branches in a planning problem’s solution space. The last modification of the example takes place in
Chapter 5, where the example is adapted to demonstrate a taxonomy of story branches that can result from an agent’s deliberation.

Lastly, in Chapter 6 I summarize the contributions from Chapters 3–5 and situate them as the sum of their parts. I also outline some areas for future work. The ideas for future work are immediate next steps or experiments that could realistically be accomplished in a short time frame.
2.1 Story Generation

As early as Schank and Abelson [SA77], it was theorized that plans could represent story plots based on the theoretical overlaps of action-oriented, causally linked, and temporal ordering of plan steps. The origins of automated story generation can be traced to the Tale-Spin system [Mee77]. Tale-Spin makes use of an inference engine to direct character action when an initial state and environment are provided. Automated planning was later used to capture author goals in the Universe system [Leb85], which focused on causal coherence of actions in the stories it generated and ensured specific author outcomes in a goal state. While other story generation mechanisms are less formal and explore less quantifiable aspects of narrative (e.g MEXICA [PS01]), I limit the scope of this work to intentional plan-based story generation methods and their more formal representation as they capture a character agent’s theory of mind and enable a direct application of previous work in classical planning.

Following the advances made by Universe, the Fabulist system [RY10] developed the Intent-driven Partial Order Causal-Link (IPOCL) plan representation. The Fabulist system leveraged the declarative representation of automated planning and generated plans where every step of the plan was in service of intended character goals, captured by the intention frame data structure. At the root of the intentional behavior modeled by IPOCL intention frames is an agent’s ability to act practically and rationally. One common approach to characterizing this behavior is the Belief-Desire-Intention (BDI) theory of mind. Beliefs are facts that an agent believes to be true; desires are goals or states a character agent wishes to
make true and may be mutually exclusive. Intentions are those desires that an agent is committed to act upon to make true and are mutually exclusive.

The concept of intention is a recent one, first introduced by Bratman and related to artificial intelligence by Cohen and Levesque [CL90], who are careful to prescribe under what conditions intentions are adopted and dropped. The model employed by IPOCL follows the adoption model by Cohen and Levesque, where once goals are adopted they are always achieved. This results in a causally connected set of character actions that support character agent believability, which is correlated with narrative comprehension. IPOCL creates these intention frames by using plan-space planning and notes that the most difficult technical limitation of generating IPOCL plans is deciding if an action is part of an intention frame. This technical limitation, along with only operationalizing the adoption of intentions (dropping goals is not addressed), has left open questions on how to explore the search space most effectively and how to extend the representing to include the dynamic nature of intentions.

Picking up where IPOCL’s representation left off from Cohen and Levesque, the Conflict Partial Order Causal Link (CPOCL) representation by Ware et al. [War14] operationalized dropped goals in story plans as conflicting character intentions. Conflicting character goals are viewed as an essential literary element of stories [Abb08; Bal97] and are represented in story plans by causal link threats. That is, one character’s intention frame foils another’s if it threatens the causal links between plan steps. CPOCL represents the remaining steps of a foiled intention frame as non-executed steps to indicate the goal was dropped. The authors validate the chosen model of conflict through two human participant studies, but acknowledge that without other conflict models with which to compare results, the results had to be compared against a naive baseline and average participant performance.

The expressivity of the CPOCL algorithm for story generation is briefly discussed by the authors, namely that equally satisfying CPOCL story plans can capture different semantic meanings, which can serve alternate authoring goals not captured in the story planning problem. Similarly to IPOCL, CPOCL does not make guarantees on the presence of conflict in a story plan, and without a model of the narratological differences between two plans, it is left to the author to characterize which POCL story plan, and linearization, offers the most desirable solution.

### 2.2 Intention Revision

The use of character intentions in narrative plan representations by IPOCL and CPOCL is based on the Belief Desire Intention (BDI) thoery of mind framework. Adopting intentions and dropping intentions both operationalize narrative theoretic concepts, specifically believable characters and character conflict, respectively. These adopt-drop operationalizations form the basis —with some additions I formalize later in this dissertation— for capturing the more dynamic nature of intentions as intention revision and what might be viewed as adopt-drop-adopt of goals. Intention revision also has some narratological concepts it can operationalize [FY17].
Algorithm 1 BDI Agent Control Loop ([Van07])

1: $B = B_0$
2: $I = I_0$
3: $\pi = \text{null}$
4: while true do
5:     get next observation $\omega$
6:     revise $B$ on the basis of $\omega$
7:     if $\text{reconsider}(B, I)$ then
8:         $D = \text{options}(B, I)$
9:         $I = \text{filter}(B, D, I)$
10:     if not $\text{sound}(\pi, B, I)$ then
11:         $\pi = \text{plan}(B, I)$
12:     if not $\text{empty}(\pi)$ then
13:         $\alpha = \text{hd}(\pi)$
14:         $\text{execute}(\alpha)$
15:         $\pi = \text{tail}(\pi)$

Within the BDI research community there has been substantial research on belief revision and update [Alc85; KM91], while there has been only cursory investigations on the cascading effects of belief changes to other mental states, namely intention. This is likely due to the still evolving and contested definition of exactly what an intention is, as well as intention’s interdependence on other BDI components. As part of the initial investigation into intention revision, Van der Hoek et al. formalized intention revision in Linear Time Logic (LTL) [Van07] based on the BDI Agent Control Loop, as specified in Algorithm 1.

More specifically, intention revision is concerned with the reconsider function (line 8) and its relationship to obtaining new observations (line 6). The reconsider function is viewed as a costly process, while new observations (a propositional or temporal formula) are obtained with relative ease, making reconsidering at every possible observation infeasible. The authors do not specify exactly when agents may reconsider, only that observation and enablement of previously un-achievable goals alone are not sufficient for a rational agent to reconsider their intentions. On the other hand, when observations are made that make a current intention un-achievable, the agent would be well served to reconsider and execute lines 9-12 to develop a new plan to an achievable goal. It is under these conditions of observations which foil current intentions that I will later define intention revision in plans.

Not only is intention revision a component of the BDI model, but it also has a prominent role in narrative. Fendt and Young [FY11] outline five areas where the dynamics of intention revision are applicable in narrative. Building on the fundamental work on character intentions [RY10], Fendt and Young developed an algorithm that is capable of initiating an intention revision on the character’s part when they receive belief updates. They extended this initial work on intention revision to a more specific
Table 2.1 QUEST questions and legal answers

<table>
<thead>
<tr>
<th>Question</th>
<th>Event</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>why</td>
<td>causal antecedents</td>
<td>superordinate goals</td>
</tr>
<tr>
<td>how</td>
<td>causal antecedents, style specifications</td>
<td>subordinate actions</td>
</tr>
<tr>
<td>enable</td>
<td>causal antecedents</td>
<td>superordinate goals-actions and subordinate actions</td>
</tr>
<tr>
<td>when</td>
<td>causal antecedents, time indices, or both</td>
<td>superordinate goals and subordinate actions</td>
</tr>
<tr>
<td>CONS</td>
<td>causal consequences, causal antecedents, or both</td>
<td>superordinate actions</td>
</tr>
</tbody>
</table>

example using suspense as the catalyst for intention revision. By combining IRIS with suspenseful templates and a Drama Manager, this complement of tools generates story plans that contain a disparity between audience and character knowledge to create suspenseful moments of dramatic irony. They create this definition of suspense and intention revision by eliminating paths to the intended goal and then forcing a new intention to be adopted. Two experimental validations were performed to evaluate the definition of suspense. The first is an author-centric evaluation to assess if IRIS would generate stories as good as humans would with the same story grammar. Test subjects were asked to read story fragments created by an experienced creative writer, IRIS, and a random baseline, then asked to rate their suspense. The results showed that IRIS generated story fragments were as suspenseful as human generated counterparts. The second evaluation measured player affect in an interactive text adventure game. Subjects were presented variants of a game testing the necessity of IRIS-generated suspenseful content. Their results showed that the game variants with suspenseful story moments were rated significantly more suspenseful than other variants. Overall, the results of these evaluations demonstrated that IRIS uses intention revision as a mechanism for creating suspenseful content. The key extension I return to later is that IRIS added the need to revise intentions when it is not possible to achieve the original intention.

2.3 Narrative Comprehension

The BDI framework forms the basis of IPOCL’s believable and rational characters. Of central importance to the validation of the IPOCL representation is the QUEST Knowledge Structure (QKS). The QKS serves as the primary information source to represent semantic content in the QUEST narrative comprehension model developed by Graesser and Franklin [GF90]. As part of the model, the QKS represents a story in a propositional manner and is manipulated to access answers to questions. The questions take the form of Why?, How?, When?, What enabled X? and What are the consequences of X?, whose legal answers are dictated by whether statements in a text were events or intentional actions (Table 2.1).
Table 2.2 QUEST node descriptions

<table>
<thead>
<tr>
<th>Node</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal</td>
<td>A state or event desired by an agent</td>
</tr>
<tr>
<td>Event</td>
<td>State change during a time period</td>
</tr>
<tr>
<td>State</td>
<td>State that is unchanged over a time period</td>
</tr>
<tr>
<td>Style</td>
<td>Articulates the manner in which an event is performed</td>
</tr>
</tbody>
</table>

Table 2.3 QUEST arc descriptions

<table>
<thead>
<tr>
<th>Arc</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consequence</td>
<td>A causes or enables B</td>
</tr>
<tr>
<td>Implies</td>
<td>A implies B</td>
</tr>
<tr>
<td>Reason</td>
<td>B is reason or motive for A</td>
</tr>
<tr>
<td>Outcome</td>
<td>B specifies whether or not the goal A is achieved</td>
</tr>
<tr>
<td>Initiate</td>
<td>A initiates or triggers the goal in B</td>
</tr>
<tr>
<td>Manner</td>
<td>B specifies the manner in which A occurs</td>
</tr>
</tbody>
</table>

QUEST is able to predict the goodness of answer (GOA) to a given question from Table 2.1 by traversing a graph representation of a story. The graph representation is comprised of three different types of nodes (goals, events, states) and six arcs (consequence, reason, outcome, initiate, implies, manner), which are described in more detail in Tables 2.2 and 2.3 respectively.

Question answer pairs can be any two nodes in a QUEST graph, and their GOA is determined by the QUEST convergence mechanism. The convergence mechanism is made up of three components. The first is the arc-search procedure that traverses types of edges in a directed manner. Second is the structural distance metric. This measures how proximal or distal a node pair is and correlates negatively to GOA. Finally, constraint satisfaction uses an average of five dimensions to arrive at a measure between zero and one indicating the answer’s ability to satisfy constraints of the the question.

While the QUEST model has been used extensively for story comprehension in generated stories, it has surprisingly only dealt with character goals that are achieved. There is no accounting for goals that are not achieved or the general dynamic nature of goal-oriented behavior.

Building on QUEST’s articulation of superordinate goal structures and their usefulness for question answering, Long et al. [LG93] found that goal-related inferences played another important role. While the QUEST model is able to accurately predict goal-related inferences made after comprehension (i.e. after the story was finished), it is not able to determine whether these inferences are generated during comprehension. Long et al. [Lon92] addressed this limitation and validated the global model of inference generation. This model says causal connections are made by readers answering Why? questions (internally) after each action and event. Generating these causal connections establishes a overall text coherence by placing an action into a belief-goal-plan chain [Sch86]. To confirm this process, subjects
read short narrative texts on a computer screen and were asked to identify whether a superordinate goal was a word or not. This lexical decision task (LDT) provided enough time (1000ms) for subjects to generate superordinate goal inferences and sufficient time for temporary activations to decay. Of interest to my work on intention revision was the resulting data that showed that inferences about super-ordinate goals are made online.

2.4 Problem Formulation

The generative nature of planning leads to a disconnect between the formulation of a domain-problem pair and the resulting structure of the solutions. Obtaining satisfying performance and results from a planner often requires analyzing the properties of the planning problem. Because of the computational expenses of planning, the problem is analyzed in advance of the actual planning process and the analysis is used to provide the planner with information to increase performance, such as promising areas of the search space.

This process of analyzing the planning problem, known as problem formulation, is of particular interest to finding plans that support the intention revision model, as the domain author is not the sole input to the planning problem, rather the exceptional actions introduced by the IN player is creating part of the problem. In fact, the problems I am concerned with are ones the domain author would hope to avoid, as they are problems that are the results of having to re-plan.

First efforts into problem formulation centered on capturing application context knowledge in a common representation [MP97]. This commitment to a common representation opened up research areas in validating whether a problem was indeed solvable with the given domain, rather than attempting to have a planning algorithm exhaustively traverse the search space. Capturing these ideals, McDermott et al. [McD98] developed the Planning Domain Description Language (PDDL), which has received several additions and expansions.

Another area that benefited from focusing on problem structure rather than the relative merits of the planning approach is search speed. Blum and Furst [BF97] developed the planning graph structure for their Graphplan planning algorithm that guarantees returning the shortest possible partial-order solution plan (if one exists).

The planning graph is a leveled graph that alternates between proposition and action layers (Figure 2.1). Each successive layer is monotonically increasing; propositions and actions are added whenever they are achieved or applicable. Construction of planning graph layers ends when either the goal propositions have been reached or the planning graph has leveled off (two successive proposition layers have identical contents). There are several variants of the planning graph (see [BK07] for an overview), whose main difference is including edges to indicate mutually exclusive actions. Once constructed, the Graphplan algorithm extracts a solution, the details of which I omit.

The impact of the development of the planning graph is far reaching. Its polynomial runtime is a
Figure 2.1 Planning graph developed by Blum and Furst [BF97]. The alternating proposition and action layers eventually reach the goal propositions in the final layer from which a solution can be extracted. The solid lines indicate the precondition/effect relationships between propositions and actions. The move L-P action in time 1 requires the preconditions at R L and fuel R be in time 1. The dotted lines represent mutually exclusive relationships with line between move L-P and the fuel R proposition at time 2 indicating it is not possible for fuel R to be true after executing move L-P.

feature attractive to heuristic search planners, which Hoffman leveraged into the development of the Fast-Forward planning system [Hof01; Hof02]. Due to the quick runtime complexity of the planning graph, Fast-Forward planning computes the planning graph after the addition of a step to a plan. In the story planning domain, Ware and Young [WY14] extended the representation of the planning graph to account for actions that were potentially motivated. That is, before actions could be added to the planning graph, they are required to be present on some character agent’s action sequence toward an active goal. Additionally, the Fast-Downward planning system [Hel06] has successfully leveraged another type of structure, the causal graph, that is computed in advance of solving the actual planning problem.

The speed advantages afforded by using the planning graph on the well-specified IPC benchmark problems are well documented. In contrast to these well-specified problems, mixed-initiative planning research strives to capture and elicit knowledge from the domain that is either unknown in advance of the planning problem or not formalized in plan-based representation. Researchers have turned to generating sets of qualitatively diverse plans to find appropriate solutions to these under-specified problems [FW16; Mye99; Ngu12; Rob14; Sri07].

Along with augmenting story plan representation to include conflict, Ware and Young [WY14] made another significant contribution to story planning in the Glaive narrative planner. Recall that one of the limitations of IPOCL planning was ensuring that when a step was added to a plan to fix a flaw, the step was part of an intention frame. Without the means to determine this quality in advance, steps are added that will not fit into an intention frame, leading to an inefficient search with many dead end plans. The state space planning community has been buoyed significantly by Hoffman’s [HN01] use of the
planning graph as a heuristic calculation to direct a state-space planner. The Glaive planner makes use of the planning graph with two augmentations. First, goal graphs are computed to determine all possible causally linked sequences of actions leading to a goal; see Figure 2.2. Computing goal graphs is a fixed polynomial cost for the planning problem. Secondly, these goal graphs are checked to ensure an action is on a path toward a character goal before the action is added to a plan graph. When actions are confirmed in the goal graph, they are termed potentially motivated. This check ensures that all actions are part of an intention frame. The goal and plan graphs enable Glaive to solve story planning problems with a very efficient search. I will both build on the plan graph for planning problem insights and leverage use of causal links later in this dissertation.

2.5 Solution Diversity

The concept of solution diversity consists of two entangled problem areas. The first is concerned with generating a diverse solution set. The domain-specific semantics of determining qualitatively different plans means domain-specific heuristics must be developed to account for them. Additionally, obtaining a diverse solution set necessitates solving a planning problem multiple times. Planning is already an expensive task and finding solutions from different areas of the search space places an increased computational burden on the planning task.

The second problem within diversity is that of evaluating diverse plan sets. Evaluating plan sets it is concerned with measures that capture the diversity within a plan set, with the primary measure being the plan set diversity [CMa11], a pairwise comparison made between every solution plan in \( \Pi \),

\[
Div(\Pi) = \frac{\sum_{\pi,\pi' \in \Pi} D(\pi, \pi')}{\frac{|\Pi| \times (|\Pi| - 1)}{2}},
\]

where \( \pi \) and \( \pi' \) are plans in a plan set, \( \Pi \), and \( D(\pi, \pi') \) is a distance metric. Additionally, the relative diversity of a plan set relative to a single plan and is defined as

\[
RelDiv(\pi, \Pi) = \frac{\sum_{\pi' \in \Pi} D(\pi, \pi')}{|\Pi|},
\]

where the symbols are the same as Equation 2.1. The difficulty in applying Equations 2.1 and 2.2 to a plan set is the definition and selection of the distance metric \( D(\pi, \pi') \). Plan-based distance metrics fall into two main categories. The first, domain-independent, are useful for comparing syntax of plans that do not contain much semantic content. Action, causal, state, compression distance, and landmark [Bry14;
Figure 2.2 The planning problem components (domain, initial state, goal state, key/objects) with corresponding problem formulation structures developed as part of the Glaive planner [War14]. The goal graph captures an agent’s subplans to achieve their intention and are used as a filter when adding actions to the plan graph.
distance metrics are all based on Jaccard distance, where the intersect of the two sets is divided by the union. A second type of distance metrics, domain-specific distance metrics, captures the semantic differences between plans. While there is less in common among these distance metrics, their foundations build on the Jaccard distance, and story plan metrics based on cognitive psychological foundations have been developed and applied [AB16; FW16].

2.6 Experience Management

Originating with Bates [Bat92], a drama manager would intelligently manipulate a virtual world that is driven by some model of experience quality. Research in areas such as graphics, camera control, non-player character (NPC) behavior, and player modeling has led to the emergence of Experience Managers (EM) to manage these mechanisms during interactive narrative gameplay. The key difference between a drama manager and EM is that not all experiences need to include drama, such as education or training, making an EM a more generalized manager but still remaining an unobtrusive presence when operating.

In lockstep with the growth of the interactive narrative community, a research community has emerged to advance the state of the art of EMs. Roberts and Isbell’s survey of drama managers [RI08] consisted of ten desiderata (Table 2.4) drawn from the author’s own DM experience, design motivations from published research, and discussions with leading researchers. Their assessment covered three classes of techniques for drama management. The first, optimization-based systems, heavily rely on machine learning to support authorial intent as an evaluative function. A second is plan-based systems that rely on symbolic representation to manipulate subjects’ knowledge structures through an action oriented paradigm. Finally, non-optimization and non-planning based systems included a range of techniques such as a hybrids of statistical and symbolic methods. Comparisons within each class were made across four common DM components. As one might expect for such a complex problem and wide time range, there was no conclusive leader within or between the classes, as each system involved trade-offs among the desiderata. Of note to my research contributions is that all three plan-based systems are listed as lacking in ease of authorship and measurement desiderata.

One area where an EM risks being obtrusive is navigating the friction created between player agency, a desirable effect of INs, and the story structure desired by an author. Mediation addresses some of these issues by striving to balance player autonomy while maintaining a coherent story arc. Breaking mediation down further, Riedl et al. [Rie03] identified two mediation strategies: intervention and accommodation.

The first, intervention, allows users to take actions, but it does not allow their effects to break the story arc. Secondly, accommodation allows the user action to break the existing story arc and transition users to a new one. While both strategies have their advantages, in this work I focus on addressing limitations of accommodation.

Automated planning has proved itself useful in the interactive storytelling research community. As
Table 2.4 Drama Manager desiderata descriptions and plan-based systems [RI08])

<table>
<thead>
<tr>
<th>Desiderata</th>
<th>Description</th>
<th>Mimesis</th>
<th>ASD</th>
<th>Dilemmas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td></td>
<td>○</td>
<td>●</td>
<td>○</td>
</tr>
<tr>
<td>Coordination</td>
<td>●</td>
<td>○</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>Replayability</td>
<td>○</td>
<td>○</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>Authorial Control</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>Player Autonomy</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>Ease of Authoring</td>
<td>○</td>
<td>○</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adaptability</td>
<td>○</td>
<td>○</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>Soundness</td>
<td>○</td>
<td>●</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Invisibility</td>
<td>○</td>
<td>○</td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>Measureability</td>
<td>○</td>
<td>○</td>
<td></td>
<td>○</td>
</tr>
</tbody>
</table>

As a result of this success, efforts have been made to create EMs to mediate subjects effectively through unintended situations [RB14; RY14a]. The common representation that they all share is a mediation tree that consists of narrative plans as nodes and edges as exceptional user actions. Traversing a mediation tree yields a linked policy for controlling actions in a story world. Research by Robertson and Young [RY14b; RY14a] operationalized Riedl and Bulitko’s concept of narrative trajectories by representing mediation as a game tree. Alternating layers of vertices representing states are connected by player and non-player character (NPC) actions managed by an EM. The EM ensures that NPC actions lead to author-specified goal states as well as a high quality experience.

At the core of ensuring this experience is a system response oracle tasked with providing optimal plans for experience. The game tree captures consistent, constituent, and exceptional actions so that the system response oracle can be engaged whenever necessary. GME is prescriptive in its outline of a system response oracle to provide an optimal experience plan; however, other than being a solution plan there is little discrimination applied to these plans. I investigate the means to provide metrics to quantify what an optimal plan might be for intentional plans.

When I define accommodation in a plan-based representation, I am concerned with user actions that break the existing story plan’s causal chain (exceptional user actions) and require the EM to perform an accommodation. The silent assumption that justifies performing accommodations is that it increases a player’s sense of agency in gameplay. Research by Ramirez and Bulitko [RB14] sought to validate this assumption by executing accommodations in a plan-based interactive narrative. Accommodation properties were driven by a player model that, as part of the player-specific automated storytelling (PAST) experience manager, could both provide an experience consistent with the player type and maintain consistency with authorial goals. The player model [Thu07] was based on the five role playing game (RPG) types (Fighter, Power Gamer, Storyteller, Method Actor, Tactitian) [Law02] and action templates were annotated with the same RPG types. When a player created a plan rupture (an action...
that invalidated authorial goals), the player’s proclivity toward an RPG type would be calculated from
their previous actions and a new story plan with actions corresponding to that RPG type generated.
This accommodation strategy is motivated by the *control heuristic* from Thomson et al. [Tho98] with
the assumption that when player actions lead to more desirable states, more agency is perceived. This
assumption and accompanying accommodation strategy were evaluated using the instrument prescribed
by Vermeulen [Ver10] for effectance and enjoyment on a choose-your-own-adventure style interactive
game based on Little Red Riding Hood. The experiments consisted of two independent variables, gaming
skill and experience management (comprised of a player model). No interaction was found between them,
but overall accommodating ruptures increased fun and agency in the study. This study, while important in
its validation of a long-held assumption, still leaves a question about the reasons accommodations do
increase agency.

2.7 Narrative Experience

An EM’s impact on a user’s experience is often difficult to measure objectively. The many components
of interactive systems result in challenging experiment designs to isolate their individual effects on a
user’s experience and to control for the same experience. This has led to the development of subjective
experience instruments that measure a user’s overall experience, rather than the impact of individual
components.

Perhaps the most widely used instruments in interactive systems are the presence questionnaire
(PQ) and the immersive tendencies questionnaire (ITQ) developed by Witmer and Singer [WS98].
The PQ is based on their definition of presence as a subject’s experience of being in one place while
actually being physically located somewhere else. They argue that presence is an awareness phenomenon
requiring directed attention and an interaction between senses that encourages a subject to become
immersed and involved in an experience. In contrast, the ITQ assesses an individual’s tendencies to
become involved in environments. Interestingly, the results from using the PQ and ITQ together show
that individual tendencies from the ITQ predict presence measured by the PQ. This relationship has led to
ITQ-PQ instruments being widely used and highly cited for interactive systems with a strong interactive
component.

In addition to the complexity of interactive systems, interactive narrative systems can add complexity
by manipulating narrative structure. Before the advent of interactive narrative, narrative text was the
predominant medium for experiencing narrative and manipulations of the structure were of interest. As
such, the *narrative transportation scale* was developed to measure the extent a reader is transported
into a story world and becomes involved with its characters. Narrative transportation itself is a concept
from Gerrig’s theory of narrative transportation [Ger93]. The TS is composed of eleven items (plus one
additional item per character in the story) from which three subscales (cognitive, affective, imagery) are
identified. While the use of subscale measures are appropriate for making claims, results from the original
validation indicate the subscales did not predict different outcomes. Perhaps most importantly, the TS scale was sensitive to manipulations of text. The authors altered a violent text (“Murder at the Mall”) to one without violence and retained much of the structure and theme, and found the TS was sensitive to the change. The TS development accounted for gender differences (where women reported higher transportation than men), distinctness from the Need for Cognition scale (non-significant correlation), and similarity to Tellegen’s Absorption Scale (moderate correlation). Since its development, the TS scale has been cited over two thousand times but surprisingly has yet to be applied to formal representations of narrative (e.g. plan-based).

Along with the transportation scale, another motivation for understanding subject experience of narrative came from the learning sciences community. Educational psychology has investigated the general concept of student interest as it relates to learning [HR06]. Interest guides children to learn and is a psychological state that emerges from a person interacting with an activity. Situational interest is an attractive psychological state that is short lived, common across individuals, and within a context. Schraw [Sch97] proposed a model that distinguishes six sources of situational interest to develop the six subscales that comprise the Sources of Interest Questionnaire (SIQ). Additionally, the authors developed the Perceived Interest Questionnaire (PIQ) to identify an overall feeling of situational interest in a text. The PIQ focuses exclusively on the reader’s assessment of their own feelings of interest while the SIQ assesses the text’s content and structure. Like the TS, the PIQ and SIQ have been widely applied but not to formal representations of narrative.

In Chapter 5, I will discuss the application of the transportation scale and perceived interest questionnaire to narrative texts generated from intentional plans.
Intention is a relatively new concept in the academic literature when compared to other mental states such as desires and beliefs that were first theorized by Socrates. This is surprising, as it seems to have the power to govern so much of our future behavior. Without knowing another agent’s intention, an observer is faced with a greater ambiguity when guessing what future actions the agent might take. In contrast, knowing an agent’s intention instantly reduces a large space of actions down to the ones that might help the agent achieve their intention (under an assumption of rational behavior). This filter of admissibility, as characterized by Bratman [Bra03], is applied automatically and instantaneously in human cognition, and with it an expectation of what actions may be observed in the future. Consequently, when an AI system needs to update a user’s filter, careful attention should be paid to a user’s mental state, so as to ensure it is setting their filter and expectations correctly.

Ascribing mental states, such as intention, to agents (artificial or not) is called a theory of mind. I focus on one theory of mind framework in particular: the Belief-Desire-Intention theory of mind (BDI) [Bra03]. It of course captures the powerful intention mental state but also two other elements necessary for developing intentions. Beliefs, simply put, are facts an agent believes to be true or false in their environment. While more advanced models assign probabilities [e.g. SP11] and can include temporal components [e.g. Van07], it is perhaps easiest to think of beliefs as true or false statements. Secondly, desires are facts in the world that an agent may wish to make true. They can be mutually exclusive (e.g. go to work, go on vacation) and remain as a desire until an agent takes action to achieve them. Once action has been taken to achieve a desire, a desire becomes an intention, and with it, an expectation of
future action. The expectation of commitment to future action comes from an agent continuing to take
take action until they achieve that intention or believe it is impossible to achieve, at which point they will drop
the intention.

Intentionality’s action- and goal-oriented nature overlaps with the basic properties of an artificial
intelligence method called partial-order, causal linked (POCL) planning [PW92]. While planning itself has
been more widely-applied to industrial applications (perhaps most well known is the Mars rover [Bre05;Lak18]), a variant called intentional planning has operationalized the overlap with BDI to generate
plans where each agent has a BDI model of the planning environment [RY10]. These intentional agents
can cooperate and conflict while pursuing their intentions and have been widely applied to
story generation systems. Using existing cognitive psychological research on human comprehension of
stories, the intentional agents generated by intentional planning systems have shown to have believable
behavior [RY10]. Specifically, the QUEST cognitive model defines a knowledge structure that represents
a reader’s mental model of the causal structure and intentional behavior created when reading any
narrative text. Additionally, QUEST is used as a part of a question-answer protocol useful for assessing
and evaluating mental models created by a text. This powerful model has been used to show believable
and goal-oriented behavior of the agents generated by IPOCL-based story generation systems [RY10].

The most widely used application of intentional planning has been as a story generation system that
provides the story content for interactive narratives (IN). The generative nature of planning enables an IN
to branch and adapt intentional (non-player) agent behavior to user input and actions. Revising a BDI
agent’s intentions to the environment has been a subject of discussion since the first concepts of intention
were introduced [Bra03]. Though the lack of an agreed upon definition of an intention has hindered
significant progress, the concept of intention revision has two key parts. The first is when an agent should
reconsider their original intention. It would be intractable for an agent to assess their intentions relative to
every new observation and belief change, as belief changes occur too frequently. Therefore, reconsidering
is reserved for those observations that would provide some utility, namely that reconsidering will likely
result in an intention revision. Second, when an agent is reconsidering an intention, an agent deliberates
on whether to keep the intention, albeit with a new set of actions to achieve it, or to drop it and replace it
with a new one.

However, the under-specified conditions of when to reconsider an intention and the lack of exact
deliberation logic of how revision takes place leads to a limitation when applying intentional planning to
interactive narrative. Concretely, when an intentional plan is executed in an IN and a user action renders
the current plan invalid, the IN requires a new plan that accounts for the intentional expectation created
in the user’s knowledge structure. To account for this limitation, I address the two intention revision
properties with plan-based definitions and an evaluation that shows the comprehension structures created
when transitioning a user to a new plan.

First, the reconsider function is activated when a causal link threat is introduced by an unexpected
action from the interactive narrative environment, be it from a user or otherwise. A causal link threat
has a formal IPOCL definition that I will visit later, however the intuition is that causal link threats make it impossible for the current intentional plan to execute to completion. Specifically, steps of an intentional plan require particular world states to be true before they can execute. When these conditions are interfered with, such as with an unexpected user action, steps may not be able to execute. A single step not executing introduces more unexpected conditions and initiates a cascading effect where more steps cannot execute, with yet more unexpected conditions and so on. In order for the story and interactive narrative to avoid a premature end, an intentional planner will generate a new story to resolve the causal link threat introduced by unexpected action. While other planning formalisms and conditions may exist with sufficient conditions to initiate a reconsidered intention, the use of a causal link threat does ensure parsimonious use of reconsidering intentions.

Second, to support an agent’s deliberation process of revising an intention once it is being reconsidered, I formalize three steps. The first step determines if an agent believes a reconsidered intention is unachievable. Adopting an intention implies a strong commitment on the agent’s part to achieve it, therefore dropping the intention is only permissible when it is unachievable. If no plan that includes the current intention can be found, the agent may adopt a new intention in its place. Second, when a new intention is identified it needs to support solving the planning problem. This is essential, as adopting a new intention that does not support solving the planning problem leaves us with another story that will come to a premature end. Finally, the goal of the new intention needs to be sufficiently different than the reconsidered intention. I accomplish this by ensuring that the new intention is not simply a sub-goal of the reconsidered intention. While it is possible that some sub-goals are sufficiently different than the original intention, ensuring it is not a sub-goal offers a higher likelihood. Together these three steps provide a balance to the deliberation of revising an agent’s intention and supporting the author’s goals embedded in a planning problem.

After I provide the details of plan-based intention revision, I move to evaluating its affect on human comprehension. I follow other intentional plan experiments on human subjects [RY10; CR16; Far16] by leveraging the QUEST cognitive model’s knowledge structure in a question-answering protocol. I define a QUEST knowledge structure subgraph designed to represent a user’s mental model after experiencing plan-based intention revision. I also identify the question-answer pairs that can validate how well the knowledge structure represents a user’s mental model.

Lastly, I evaluate the agreement between my QUEST knowledge structure subgraph predictions and user responses to my plan-based model of intention revision. My human subject experiment findings indicate that my QUEST knowledge structure subgraph represents a user’s mental model of plan-based intention revision, as hypothesized.
3.1 Intentional Planning

Intentional planning’s goal is to generate believable action sequences for the agents in a plan. This is done by ensuring an agent’s actions support achieving that agent’s intention. I use definitions for intentional planning based on Riedl and Young’s work on IPOCL [RY10]. Intentional planning differs from classical planning with one key additional constraint on the solutions: all steps in a solution plan must be causally linked to achieving at least one agent’s intention (happenings are fate’s intention). This causally linked set of actions is referred to as an agent’s sub-plan to achieve their intention. An agent, their sub-plan and intention are all captured in an aggregate structure called an intention frame that is included in the definition of an intentional plan. Below, I formally define the above intentional planning concepts and provide concrete examples using a simple intentional plan and planning problem.

I use a simple example throughout this chapter (and others) to make salient points about the contributions my work makes to interactive narrative. The example characterizes the intention revisions non-player characters make in response to a user when using an intentional plan to structure the story content. The planning problem is set in a prison where a non-player character, a prisoner called Smith, begins in jail and has desires to get out of the jail. Smith is an intentional agent; that is, he forms intentions by taking actions to achieve his goals. Another agent, a user agent called the warden, represents the control a user has in an interactive narrative. To begin, I formally define a planning problem:

Definition 1 (Planning problem) A planning problem $\Phi$ is a five-tuple $(I, G, O, A, \Lambda)$ where $I$ and $G$ are conjunctions of true literals in the initial state and goal conditions respectively, $O$ is the set of symbols referring to objects, $A$ the set of symbols referring to agents, and $\Lambda$ a set of action schema.

An object in a planning problem is a uniquely identified, typed symbol. In the example in Figure 3.2, there is a lone object, the money object that is of type Asset. The agents, Smith and warden, are also technically objects, but I separate them out for semantic reasons, as they are the only objects with intentions. Relationships between objects are specified using predicates and are often referred to as ground literals. For example, in the initial state, the $\text{has(warden, money)}$ predicate identifies who has the money.

The initial state and goal conditions provide a domain author (the individual who creates the planning problem) the opportunity to declare the true predicates when an IN begins and the predicates that are required to be true for an ending condition. In Figure 3.1, Smith begins in jail, $\text{inJail(\text{Smith})}$ in $I(\Phi)$, and the only authorial goal in $G(\Phi)$ is $\text{¬inJail(\text{Smith})}$. By abstractly specifying the goal condition, rather than specific agent goals (e.g. $\text{exonerated(\text{Smith})}$), it allows Smith the flexibility to revise his intentions and still support solving the planning problem.

Intentional planning uses agents and the actions they take to achieve their goals as the means to solve planning problems. The Prison example has two intentional agents, Smith and the warden.

Definition 2 (Agent) A symbol that uniquely identifies a goal-oriented agent in a planning problem.
Both Smith and the warden are defined, along with their type, in the Agents of planning problem ($\mathcal{A}(\Phi)$) in Figure 3.1. An agent’s type restricts the actions it can execute. For example, the ApproveTrial action requires an agent of type User to execute it, and therefore Smith does not consider it as an option when pursing his goal, as he is of type Prisoner.

An intentional planner generates plans that contain action sequences to support agents pursuing their goals.

**Definition 3 (Agent Goal)** An agent goal is a logical sentence of literals that identifies a desired world-state of an agent.

Agent goals are represented by the intends (agent, goal) operator. In the example, Smith takes action in pursuit of his goal of being exonerated ($\text{intends (Smith, exonerated(Smith))}$) while the warden takes an action to fulfill his goal of helping Smith, $\text{intends (warden, hasTrial(Smith))}$. In definition 12 I will explain how goals are aggregated with other planning elements into an intention. Note that in the initial
Prison intentional plan $\{\pi\}$ including intention frames, $\mathcal{I}(\pi)$

state in Fig. 3.1, there are seemingly contradictory goals; intends (Smith, exonerated(Smith)) and intends (Smith, escaped(Smith)). While an intentional agent cannot be committed to contradictory goals [CL90], these yet-to-be-acted-upon goals are the intentional planning’s representation of desires, part of the BDI theory of mind.

The agent who executes an action is called the consenting agent. In Figure 3.2, Smith executes the MakeFriends, Embezzle, RequestTrial, and Testify actions. This is reflected in my Action definition:

**Definition 4 (Action)** Action $A$ consists of preconditions that must be satisfied before execution, $\text{PRE}(A)$, resulting effects, $\text{EFF}(A)$, and a consenting agent, $\text{AGENT}(A)$, who performs the action. Preconditions are literals in a state space whose conjunction must evaluate to true immediately before an action’s execution. An action’s effects are literals whose conjunction evaluates to true immediately after $A$ is executed.

An action’s name, parameter list, preconditions, effects, and consenting agent describe an action schema. The action schemata of a planning problem (see $\Lambda$ in Figure 3.1) can be instantiated into plan steps by grounding the free variables and result in the plan steps $s_1$ – $s_5$ in Figure 3.2.

**Definition 5 (Step)** A step is a ground instance of an action, one where all the action’s parameters are bound to the objects defined in the planning problem.

An intentional agent’s goal-oriented actions (steps) are generated within the context of an intentional plan.

**Definition 6 (Intentional plan)** An intentional plan $\pi$ is $\langle S, B, O, L, I \rangle$ where the set of steps is $S$, $B$ is the set of binding constraints on the variables of $S$, $O$ is the partial ordering of the steps in $S$, $L$ is the set of causal links joining the steps in $S$, and finally $I$ is the intention frame set that structures agents, their goals, and sub-plans.

The example *Prison* story is represented in Figure 3.2 as an intentional plan, $\pi$. Within $S(\pi)$ there are always two additional steps to those that agents consent to. They are referred to as dummy steps, as the planner does not need to generate them as part of the search for a plan nor are they required to be defined as actions in the action schemata. The first dummy step represents the initial conditions of the planning problem as a step with no preconditions and only effects.

26
Definition 7 (Start step) A start step of an intentional plan, $s_0(\pi)$ is a step with no preconditions and effects that are the same as the initial state of the planning problem. The start step is always executed before all other steps, so as to provide the initial state of the planning problem.

A second dummy step, the goal step, represents the goal conditions of the planning problem as a step that has only preconditions and no effects. Once the goal step’s preconditions are satisfied, also known as the action being applicable, the planning problem is solved.

Definition 8 (Goal step) A goal step of an intentional plan, $s_g(\pi)$, is a step with preconditions equal to the goal conditions of a planning problem and no effects. A problem’s goal step can be thought of as the last step of every plan for that problem. The goal step is always executed after all other steps, indicating the goal conditions of the planning problem have been reached.

Intentional plans use causal links to record that an effect $p$ of an action $a$ satisfies action $b$’s precondition $u$.

Definition 9 (Causal links) A causal link, $s \xrightarrow{p} u$, is a tuple $\langle s, p, u \rangle$ where $s, u$ are actions and $p$ is a literal. A causal link records that $p$ is both an effect of $s$ and satisfies the precondition in $u$.

In the example, the first step of the plan ($s_1$) has a precondition of $\text{inJail}(\text{Smith})$ that is satisfied by the start step ($s_0$). This is indicated by an edge between $s_0$ and $s_1$ labeled with the $\text{inJail}(\text{Smith})$ predicate. All the edges between plan steps in Figure 3.2 are causal links.

In addition to causal links imposing the ordering of plan steps, ordering constraints indicate the partial order of plan steps. Ordering constraints resolve causal link threats, which I define in Definition 19, by specifying whether one step must be executed before another.

Definition 10 (Ordering constraint) An ordering constraint is of the form $w \prec s$ and is interpreted as $w$ must be executed at some time before $s$.

Establishing causal links between steps requires maintaining which precondition and effect variable pairs must unify. Additionally, it may be desired that the free variables in action parameters are not equivalent. Binding constraints are used to capture both these conditions.

Definition 11 (Binding constraint) A binding constraint is a pair of variables $(u, v)$ or negated pair $\neg(u, v)$ where the pair must unify or not be allowed to unify, respectively.

The final element of an intentional plan is the intention frames. Intention frames structure intentional plan elements into goal-oriented behavior of individual agents. They are the unique representation difference between IPOCL (intentional plans) and POCL plans.

Definition 12 (Intention Frame) An intention frame is a tuple $I = \langle a, g, m, \sigma, T \rangle$ where $a$ is an agent, $g$ is $a$’s goal, motivating step $m \in S(\pi)$ with the effect $\neg g$, the satisfying step $\sigma \in S(\pi)$ with $g$ as an effect, and $a$’s subplan to achieve $g$ is the set of steps $T$. 
Figure 3.2 shows the two intention frames for Smith and warden. Note that the motivating step came from $s_0$, the start step of the plan.

Recall that desires become intentions when a commitment to act is made by the agent. To begin acting upon these desires and turn them into an intention, an agent goal needs to be motivated. When the goal’s negation is true, it spurs an agent into action. The planning community uses the closed world assumption, all facts in the world are assumed to be false unless otherwise indicated. In the example, both Smith’s goals (intends (Smith, exonerated(Smith)) and intends (Smith, escaped(Smith))) are motivated in the initial states as it is implied through the closed world assumption that $\neg$exonerated(Smith) and $\neg$escaped(Smith) are both true.

### Definition 13 (Motivating Step)

A motivating step for agent $a$ and goal $g$ is a plan step whose effects contain the motivation $\neg g$. A motivating step describes how a character came to adopt a goal.

When an agent goal is motivated, the agent may take an action toward achieving it. This commitment to action changes a desire to an intention. The intends (Smith, exonerated(Smith)) desire becomes an intention when Smith executes an action toward achieving it, the MakeFriends action ($S_1$ in Figure 3.2).

An agent’s sub-plan is a set of plan steps that are causally linked and that the agent consents to.

### Definition 14 (Sub-plan)

A sub-plan for $a$ to achieve $g$ is a set of steps $T \subseteq S(\pi)$ that $a$ consents to, where each step shares at least one causal link (in $L(\pi)$) to another step in $T$, and $\sigma$ achieves $g$. Steps in $T$ occur after $m$ and before $\sigma$.

Additionally, the satisfying step in each intention frame is ApproveTrial and Testify. I define the satisfying step as:

### Definition 15 (Satisfying Step)

A satisfying step, $\sigma$, for an agent $a$ and goal $g$ is a step that $a$ consents to and $g \in Eff(\sigma)$. It is the final step of an agent achieving a goal in a subplan.

Finally, intentional plans are generated to solve planning problems. The intentional plan in Figure 3.2 solves the problem defined in Figure 3.1 and is referred to as a solution plan.

### Definition 16 (Solution plan)

An intentional plan $\pi$ is a solution plan to the planning problem $\Phi$ if the steps of $\pi$, $S(\pi)$ have no open preconditions and $\pi$ is consistent. An intentional plan is consistent if there are no cycles in the ordering constraints, $O(\pi)$, and no causal link threats remain between two steps in $S(\pi)$.

### 3.2 Intention Revision

Building on the long history of capturing intentionality in artificial intelligence, intentional planning’s representation has operationalized concepts from the BDI theory of mind to create goal-driven character
agents with explainable behavior. From this representation, two classes of agents have emerged. The first is a ‘conscious rationalizer’ agent who can justify adopting an intention with an air-tight, causally linked sequence of actions. A second ‘fairweather’ agent drops their intention the moment a causal link threat is introduced. These agents have been applied in several story generation contexts both in interactive and non-interactive environments.

While adopting and dropping intentions is essential for goal-oriented agents, restricting agents to these two classes does not capture the dynamic nature of intentions. Intention revision can represent explainable behavior changes, which is especially important for the complex interactions between character intentions in interactive narratives, namely when we need to change behavior in response to unexpected events.

To address this limitation, I define an intention revision model with two parts. I first prescribe a case for when agents should reconsider existing intentions with logic from BDI agent design. Second, I define how agents deliberate to revise intentions using persistent goals (P-GOAL) as characterized by Cohen and Levesque [CL90]. I continue with the Prison example to demonstrate their applicability to addressing issues during the execution of an intentional plan-based interactive narrative.

During the execution of an interactive narrative with an intentional plan as a story structure, a step is labeled as executed once its effects have been updated in the execution state.

**Definition 17 (Execution State)** The execution state of a plan, $\text{exec}_i(\pi)$, is a set of consistent, non-modal, ground literals that are true after executing $s_i$, where $s_i \in S(\pi)$.

The execution state after adding the effects of each step of $\pi$ is in Figure 3.4. I use executed steps to determine active intentions and their relation to desires and completed intentions.

**Definition 18 (Active Intention)** An active intention is part of the current plan, $i \in I(\pi)$, where at least one step of the sub-plan, $T(i)$, is executed and the satisfying step, $\sigma(i)$, is not executed. The active intentions of a plan are indicated by $I^a(\pi)$.

In Figure 3.2, Smith’s intention of $\text{exonerated}(Smith)$ is active from $s_1$ to $s_4$, until he executes the satisfying step, $\text{Testify}$, at $s_5$. Active intentions are useful for identifying reconsidered intentions and
Figure 3.4 Execution state of $\pi$ after executing each step, $s \in S(\pi)$

support my definition of intention revision. During execution of my plan-based IN, the user agent (the warden) can take actions that introduce causal link threats that prevent a non-player agent from achieving their goal.

**Definition 19 (Causal link threat)** A causal link threat occurs when a causal link is established $s \xrightarrow{p} u$, and some other step $w$ has effect $\neg p$ and could be executed after $s$ but before $u$. Executing $w$ in this interval means the precondition $q$ of $u$ is no longer satisfied by $s$ and $u$ will not execute.

In Figure 3.3, the player agent executes the *DenyTrial* step ($s'_4$) instead of the planned *ApproveTrial*. The *DenyTrial* step introduces a causal link threat $\neg hasTrial(Smith)$ to the *Testify* step, negating the established precondition and making it not possible to execute to completion. I refer to an action that introduces a causal link threat at execution time as an exceptional action.

**Definition 20 (Exceptional Action)** An exceptional action $s'_t$ executed at time $t$ by the user agent, $\text{AGENT}(s'_t) = \text{user}$, where one of its effects $e \in \text{EFF}(s)$, introduces a causal link threat to a precondition of a future step, $\text{PRE}(s_u)$, in the current plan $\pi$ where $t \leq u$.

Since the *Testify* step is part of Smith’s sub-plan to achieve his *exonerate(Smith)* goal, Smith can no longer achieve this goal as planned. Furthermore, the intentional plan can no longer reach the goal conditions and will be forced to end prematurely. Recall that the reconsider function in the BDI agent control loop is intended to be used sparingly, and the argument here is that the introduction of a causal link threat at execution time meets the sufficient conditions for an agent to reconsider their intention. More generally, every causal link threat will initiate at least one reconsidered intention. A causal link threat threatens at least one step and every step in an intentional plan is part of at least one agent subplan.
In the example, the *DenyTrial* exceptional action creates a causal link threat to Smith’s subplan to achieve *exonerate(Smith)* because the subplan contains the *Testify* step. This initiates Smith reconsidering his *exonerate(Smith)* intention.

**Definition 21 (Reconsidered Intention)** A reconsidered intention, \( I_R \), is a tuple \( \langle I, \chi, \varepsilon \rangle \), where \( I \) is an active intention and \( \chi \) is the exceptional action with an effect \( \varepsilon \) that introduces a causal link threat to two linked steps in the sub-plan, \( T(I) \).

Having Smith reconsider his *exonerate(Smith)* intention is only the first part of intention revision. While causal link threats are a first example of when to reconsider an intention, they are probably not the only ones\(^1\). However, I now move to the three steps of deliberation an agent will perform as part of the intention revision. First, in order to drop an intention and revise it to a new one, an agent will deliberate whether the goal is worth pursuing. Cohen and Levesque [CL90] prescribe that an agent should only drop a goal after achieving it or when the agent believes the goal is unachievable. I focus exclusively on active intentions, unachieved by definition, and seek to determine if a goal is unachievable.

**Definition 22 (Unachievable Goal)** A goal is unachievable, \( g_u \), if no sub-plan to achieve \( g(I_R) \) exists when using a agent’s belief state as the initial state.

In my plan-based representation, agents maintain a belief state of their environment, represented as sets of consistent, non-modal, ground literals. Agents update their belief state by observing the effects of actions. After the *DenyTrial* action, Smith believes exoneration is unachievable, as no sub-plan exists to achieve exoneration. This belief leads Smith to drop this intention.

A second step to an agent’s deliberation is to find some new goal that is achievable, by way of a sub-plan, and adopt it as an intention. Finding a new sub-plan is not done in isolation and involves solving a new planning problem derived from the original one, where the initial state is replaced with the execution state.

**Definition 23 (Achievable Goal)** A goal is achievable, \( g \), if a sub-plan to achieve \( g \) exists when using a agent’s belief state as the initial state.

In the example, Smith adopts the achievable goal *escaped(Smith)* and sub-plan \( (s'_1, \text{Escape in } \pi') \) that also solves the planning problem by reaching the goal condition of the planning problem, that is, \( \neg \text{inJail(Smith)} \).

Finally, the third and final step to an agent’s deliberation is the requirement that the new achievable goal is not a sub-goal of the original unachievable goal. While this difference may seem pedantic, it

---

\(^1\)Another example of when an agent would revise their intention is when they learn that another agent has the same intention of obtaining a finite resource. They may choose to drop it entirely if they are unlikely to achieved it or adopt a subgoal as a kind of mitigation. Shared intentions offer another intriguing possibility. An agent may revise their commitment to the shared intention once they have accomplished some alterior intention or goal. This leaves the other agent to revise the shared intention to a individual one and perhaps exact revenge on the other agent.
would be possible for an agent to develop a subplan to achieve an effect of the pen-ultimate step of the original intention’s subplan. This presents some ambiguity, as while there is a difference of one action, the ultimate intention is arguably the same. For this reason, I specify that adopting a sub-goal of the original intention does not constitute intention revision.

**Definition 24 (Sub-goal)** A goal $g_2(I_2)$ is a sub-goal of $g_1(I_1)$, $\text{SUB}(g_2, g_1)$, if the intentions belong to the same agent, $a(I_1) = a(I_2)$, and $g_2$ is an effect in the subplan to achieve $g_1$, $g_2 \in \text{EFF}(T(I_1))$.

In the example, not only is the new goal $\text{escaped}(Smith)$ semantically very different from $\text{exonerated}(Smith)$, but $\text{escaped}(Smith)$ is not an effect of any action in the subplan to achieve $\text{exonerated}(Smith)$. Finally, I bring the reconsideration and deliberation together into a plan-based definition of intention revision.

**Definition 25 (Intention Revision)** An intention revision is $\langle I_R, I' \rangle$ where $I_R$ is an active intention in $\pi$ and $g(I_R)$ is unachievable, $I'$ is an intention frame in $\pi'$ where $g(I')$ is achievable and not a subgoal $\neg \text{SUB}(g(I'), g(I_R))$, and $\text{AGENT}(I') = \text{AGENT}(I_R)$.

In Figure 3.3, Smith’s intention revision from $\text{exonerated}(Smith)$ to $\text{escaped}(Smith)$ is caused by the user’s $\text{DenyTrial}$ action. I generate a new intentional plan, $\pi'$ in response, with a single action. For a summary of the relationship between the different intention revision components, see Algorithm 2. This model of intention revision leverages the powerful concept of intention to account for a user’s existing mental model of the narrative, and serves as my branching mechanism to accommodate exceptional user actions.

### 3.3 Intention Revision Comprehension

The theoretical overlap between partial-order, causal link plan representation and intentionality is an auspicious one, as it also offers some major benefits for evaluation. There is a substantial amount
of narrative comprehension research in cognitive psychology that seeks to evaluate the causal and intentional structure of stories [GF90; Gra91; Gra94; Lon92; LG93; Tra82; TB85; TS85]. This body of research is primarily focused in narrative text with some work on film [e.g. Mag05] and with more recent experimental results demonstrating the applicability of the QUEST cognitive model to human comprehension of intentional plans.

However, in these evaluations there is limitation when it comes to intention revision. No texts that were used in the original QUEST design — nor was I able find any since — had a character revise their intention over the course of a narrative. This limitation is evident in the primitives of the QUEST knowledge structure (QKS), a graph structure that represents a reader’s mental model created by the intentional and causal structure of a narrative text. The QKS has no specification or definition of edge or node types to indicate dropping, modifying, or revising intentions and goals. This representational gap poses a problem to evaluate human comprehension of plan-based intention revision. Without a strong cognitive psychological foundation for comprehension, I will lack an understanding of the effects of using intention revision as a story branching mechanism.

Rather than attempting to modify a robust and well-established cognitive model by introducing new edge or node types to capture intention revision, I aggregate existing QKS primitives to form a QKS subgraph. This subgraph captures the introduction of a causal link threat by an exceptional action (reconsidering an intention), along with the three deliberation criteria defined in Section 3.2. This approach has two major benefits. First, it can be evaluated immediately. The QUEST question-answer composition rules are prescriptive about what nodes constitute good/poor goodness of answer pairs, and I can use these as is to test the existence of the QKS subgraph. Secondly, should the QKS subgraph be an accurate representation of a user’s mental model, it provides a robust cognitive psychological foundation of comprehension for using intention revision as a strategy for accommodating unexpected user actions. The current state of the art is to create a story branch by introducing a new character (often referred to as the ‘remember-the-new-guy’ trope) or by changing existing character intentions in an unprincipled way.

Recall from Section 3.2 that when an exceptional action introduces a causal link threat to an existing intention’s sub-plan, it causes the intention to be reconsidered. In my QKS intention revision subgraph, Figure 3.5, the reconsidered intention and exceptional action are represented by the goal and event nodes G1 and E1, respectively. The Jail example is shown as a QKS intention revision subgraph in Figure 3.6 where the G1 node is Smith’s original exonerated(Smith) goal and the user’s (warden) DenyTrial exceptional action is represented in the E1 node.

To represent part of the agent’s deliberation, I add two more nodes to my QKS intention revision subgraph. The first, S1, represents the reconsidered intention’s transition to an unachievable goal. For Smith, his goal of exonerated(Smith) becomes unachievable when no sub-plan is found. The newly adopted goal is represented by a second node, G2. Smith adopts his escaped(Smith) goal in G2 when his original goal is unachievable.

The QUEST knowledge structure prescribe composition rules that must be followed when introduc-
ing edges between QKS nodes. I introduce two edges to complete the QKS intention revision subgraph. The first edge, an event-state consequence edge, represents that as a consequence of the exceptional action in E1, the original goal becomes unachievable in S1. As a consequence of the user’s DenyTrial exceptional action, Smith’s goal of exonerated(Smith) becomes unachievable. Second, the initiates edge connects the exceptional action in E1 to the new goal in G2 and is an event-goal edge. This edge links the DenyTrial exceptional action as the trigger to Smith adopting his escaped(Smith) goal. Together these four nodes and two edges make up the QKS intention-revision subgraph.

To test whether the QKS intention revision subgraph is structured as hypothesized, I turn to using QUEST in a question-answering protocol. To validate the existence of pathways between two nodes in a QKS structure, I first generate question-answer text. The question-answer protocol labels one node as the Question node and uses a template based on the node’s type to form a plain English question. Similarly, the remaining node in the pair is labeled as the Answer node and the answer statement is generated using the node’s type. For example, the user’s exceptional action in E1, DenyTrial, and Smith’s new goal of escaped(Smith) in S1 form a question-answer pair, respectively, and their generated text is captured in Q1 of Table 3.2.

A second part to the QUEST question-answer protocol is to predict subject responses to the question-answer pairs. QUEST defines a QKS graph traversal to predict a question-answer pair’s goodness of answer. Comparing the agreement between QUEST’s goodness of answer predictions and subject responses to the generated text of the question-answer pairs can be used to assess the existence, or lack, of an edge type between two nodes. In the aforementioned Q1, the QUEST traversal predicts a Good
Figure 3.6 The QUEST knowledge structure intention revision subgraph of the *Prison* example.

Table 3.2 Questions to evaluate the QKS intention revision subgraph

<table>
<thead>
<tr>
<th>#</th>
<th>Pair</th>
<th>Prediction</th>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>E1,S1</td>
<td>Good</td>
<td>What is a consequence of the warden denying the trial?</td>
<td>That Smith believes exoneration is unachievable.</td>
</tr>
<tr>
<td>Q2</td>
<td>G2,E1</td>
<td>Good</td>
<td>Why did Smith want to escape?</td>
<td>Because the warden denied his trial.</td>
</tr>
<tr>
<td>Q3</td>
<td>G2,G1</td>
<td>Bad</td>
<td>Why did Smith want to escape?</td>
<td>Because Smith wanted to be exonerated.</td>
</tr>
</tbody>
</table>

goodness of answer (GOA) due to the *Consequence* edge that connects them. The predictions for all three questions can be seen in column three of Table 3.2.

My QKS intention revision subgraph can be tested as a representation of a user’s mental model using three question-answer pairs, whose details are captured in Table 3.1. Both Q1 and Q2 assess whether the exceptional action had the desired effects of dropping one goal and adding another, representing steps 1 and 2 of an agent’s deliberation. On the other hand, I ask Q3 to ensure subjects are in fact organizing $g(I)$ and $g(I')$ into separate local goal hierarchies, representing step 3 of an agent’s deliberation. Recall in Section 3.2 that my goal is to ensure that the new goal in $G_2$ is not a subgoal of $G_1$. To ensure no hierarchical semantic relationship exists between these two goals, I question subjects on the existence of a subgoal relationship. With subject and QUEST prediction agreement on the three questions in Table 3.1, I can validate the comprehension of the plan structure of an agent’s deliberation during intention revision.

3.4 Evaluation Method

The goal of my plan-based model of intention revision is to provide a principled method to change the behavior of intentional agents. The two-part model begins with an exceptional action that introduces a causal link threat. This initiates a reconsidered intention on the part of an agent. Once an intention is
being reconsidered, an agent’s deliberation process begins. This deliberation process is designed to be interpretable by human subjects, so as to provide a justification for the intention revision. I hypothesize that my intention revision model creates a change in a user’s mental model that can be represented with the QKS intention revision subgraph in Figure 3.5. The following experiment uses three questions to evaluate whether the exceptional action does in fact lead to the desired knowledge structure and is designed to answer the following research question:

R1. How does an exceptional action affect the QKS intention revision subgraph?

### 3.4.1 Subjects

Data were collected from 192 participants who were recruited from the CrowdFlower crowd sourcing platform and paid $0.25 USD to read one narrative text and answer ten QUEST generated question-answer pairs using a four-point Likert scale.
I capture the measurement changes I expect to observe in experimental groups in the following hypotheses:

H1. Subjects who do not observe the exceptional action will rate the Q1 question-answer pair with a low GOA.

H2. Subjects who do not observe the exceptional action will rate the Q2 question-answer pair with a low GOA.

H3. All subjects will rate the Q3 question-answer pair with a low GOA.

### 3.4.2 Materials

As I previously discussed, narrative texts from earlier experiments did not include any intention revisions, thus no previous materials could be used. To develop suitable materials, I used the design criteria in Table 3.3 to create two narrative texts adapted from films to serve as the materials for the experiment.

The first criteria was to minimize the number of characters and locations. According to the event-index situation model [ZR98] (EISM), characters and locations cause event segmentation and reorganizing of mental models. To mitigate any effects of event segmentation, the stories are in a single location with two characters. Second, I explicitly state what each character’s intentions are. This is done to ensure that the correct goals and causal structures are salient in the reader’s mental model. Thirdly, I use a story statement as a proxy for an exceptional user action. This statement is designed to initiate a reconsidered intention on the agent’s part that is then integrated into the reader’s knowledge structure. A fourth criteria that opposes the third is a story statement that is not causally connected to the story. This allows us to form experiment groups that assess whether the exceptional action causes a user to form a mental model that is represented by the QKS intention revision subgraph. Finally, after the exceptional action, I include a statement that explicitly states the character is pursuing a new goal. Again, this is done so I have flexibility in creating my experimental groups.

In addition to the design criteria, the two texts were similar in causal structure, identical in length, had a single protagonist, and a single antagonist. The texts of each story are in Tables 3.4 and 3.5, their complete QKS structures in Figures 3.8 and 3.10, and their respective QKS intention revision subgraphs in Figures 3.9 and 3.11. Note that the *Prison Sentence* story in Figure 3.8 is a longer, more detailed version of the *Prison* intentional plan example I have been using. From these two texts, variants for each experimental group were derived and are discussed in Section 3.4.3.

### 3.4.3 Procedures

Upon logging into the online experiment, participants were randomly assigned to either the control or treatment groups, and then assigned to read either the *Prison Sentence* or *Farm Boy* story. The control group read a variation of the story where they received the exceptional action in statement 12a and an
Table 3.3 Narrative text design elements used to create experiment materials

<table>
<thead>
<tr>
<th>Story feature</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single location</td>
<td>Limit event effects as defined by the EISM [ZR98]</td>
</tr>
<tr>
<td>Goal deliberation</td>
<td>Provide the subject with the superordinate goal of two goal hierarchies</td>
</tr>
<tr>
<td>Relevant belief update</td>
<td>Initiate dropped ( G ) inference</td>
</tr>
<tr>
<td>Irrelevant belief update</td>
<td>Validate if any belief update would initiate inferences about ( G )</td>
</tr>
<tr>
<td>Explicit intention of ( G' )</td>
<td>Initiate pursuit of ( G' ) inference</td>
</tr>
</tbody>
</table>

Table 3.4 Prison Sentence text

<table>
<thead>
<tr>
<th>#</th>
<th>Group</th>
<th>Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>All</td>
<td>Smith is a wrongfully convicted accountant and wants to be exonerated.</td>
</tr>
<tr>
<td>2</td>
<td>All</td>
<td>Smith finds a getaway plan to Canada in his jail cell.</td>
</tr>
<tr>
<td>3</td>
<td>All</td>
<td>Smith discards the getaway plan.</td>
</tr>
<tr>
<td>4</td>
<td>All</td>
<td>Smith is always compliant with the guards.</td>
</tr>
<tr>
<td>5</td>
<td>All</td>
<td>Smith does the guards’ taxes.</td>
</tr>
<tr>
<td>6</td>
<td>All</td>
<td>Smith is always compliant with the warden.</td>
</tr>
<tr>
<td>7</td>
<td>All</td>
<td>Smith does the warden’s taxes.</td>
</tr>
<tr>
<td>8</td>
<td>All</td>
<td>Smith is forced by the warden to embezzle prison funds into a fake bank account.</td>
</tr>
<tr>
<td>9</td>
<td>All</td>
<td>Smith makes friends with some inmates.</td>
</tr>
<tr>
<td>10</td>
<td>All</td>
<td>Smith learns an inmate has evidence that exonerates him.</td>
</tr>
<tr>
<td>11</td>
<td>All</td>
<td>Smith asks the warden to help him get exonerated.</td>
</tr>
<tr>
<td>12a</td>
<td>C, T1</td>
<td>Smith is told by the warden he will never allow him to be exonerated.</td>
</tr>
<tr>
<td>12b</td>
<td>T3</td>
<td>Smith passes lots of time in prison by drawing and sketching.</td>
</tr>
<tr>
<td>13</td>
<td>C</td>
<td>Smith retrieves the getaway plan to Canada.</td>
</tr>
<tr>
<td>14</td>
<td>All</td>
<td>Smith hangs some artwork in his cell.</td>
</tr>
<tr>
<td>15</td>
<td>All</td>
<td>Smith takes a spoon from the cafeteria.</td>
</tr>
<tr>
<td>16</td>
<td>All</td>
<td>Smith digs a tunnel with the spoon behind the artwork.</td>
</tr>
<tr>
<td>17</td>
<td>All</td>
<td>Smith steals the warden’s spare suit.</td>
</tr>
<tr>
<td>18</td>
<td>All</td>
<td>Smith crawls through the tunnel.</td>
</tr>
<tr>
<td>19</td>
<td>All</td>
<td>Smith changes out of his prison clothes into the warden’s suit.</td>
</tr>
<tr>
<td>20</td>
<td>All</td>
<td>Smith withdraws money from the fake bank account.</td>
</tr>
<tr>
<td>21</td>
<td>All</td>
<td>Smith buys a plane ticket to Canada.</td>
</tr>
</tbody>
</table>
Table 3.5 Farm Boy text

<table>
<thead>
<tr>
<th>#</th>
<th>Group</th>
<th>Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>All</td>
<td>Hank works at his farm so he can sell his crops in the fall.</td>
</tr>
<tr>
<td>2</td>
<td>All</td>
<td>Hank receives an email about a hacker revolution against the government.</td>
</tr>
<tr>
<td>3</td>
<td>All</td>
<td>Hank marks the email as spam.</td>
</tr>
<tr>
<td>4</td>
<td>All</td>
<td>Hank tends his fields.</td>
</tr>
<tr>
<td>5</td>
<td>All</td>
<td>Hank plants some crops.</td>
</tr>
<tr>
<td>6</td>
<td>All</td>
<td>Hank tends his livestock.</td>
</tr>
<tr>
<td>7</td>
<td>All</td>
<td>Hank builds a barn.</td>
</tr>
<tr>
<td>8</td>
<td>All</td>
<td>Hank is forced by some government soldiers to pay them a safety tax.</td>
</tr>
<tr>
<td>9</td>
<td>All</td>
<td>Hank attends a community meeting.</td>
</tr>
<tr>
<td>10</td>
<td>All</td>
<td>Hank learns the safety tax is fake.</td>
</tr>
<tr>
<td>11</td>
<td>All</td>
<td>Hank asks for a receipt from the soldiers.</td>
</tr>
<tr>
<td>12a</td>
<td>C, T1</td>
<td>Hank’s farm is pillaged by the government soldiers.</td>
</tr>
<tr>
<td>12b</td>
<td>T3</td>
<td>Hank spends his free time watching cat videos online.</td>
</tr>
<tr>
<td>13</td>
<td>C</td>
<td>Hank retrieves the instructions from the hacker revolution email.</td>
</tr>
<tr>
<td>14</td>
<td>All</td>
<td>Hank orders computer equipment.</td>
</tr>
<tr>
<td>15</td>
<td>All</td>
<td>Hank takes computer training courses.</td>
</tr>
<tr>
<td>16</td>
<td>All</td>
<td>Hank sets up computer hacking infrastructure.</td>
</tr>
<tr>
<td>17</td>
<td>All</td>
<td>Hank gets government network blueprints from the hacker revolution.</td>
</tr>
<tr>
<td>18</td>
<td>All</td>
<td>Hank analyzes the network blueprints.</td>
</tr>
<tr>
<td>19</td>
<td>All</td>
<td>Hank designs a network destruction exploit.</td>
</tr>
<tr>
<td>20</td>
<td>All</td>
<td>Hank hacks into the government’s computer network.</td>
</tr>
<tr>
<td>21</td>
<td>All</td>
<td>Hank uses the network destruction exploit.</td>
</tr>
</tbody>
</table>
explicit statement about the revised goal in statement 13 in Figures 3.4 and 3.5. Conversely, the treatment groups read stories without statement 13 and received statement 12a (T1), no statement 12 (T2), and statement 12b instead of 12a (T3). See Figure 3.7 for the treatment groups mapped to story structure.

Participants began the experiment by completing a brief demographic survey and then read their assigned story, presented as a single page of text. After reading their respective stories, participants were provided an example of how to respond to QUEST question-answer pairs. Subjects then responded to ten randomly ordered QUEST question-answer pairs that included the three questions (Table 3.1) needed to test the QKS intention revision subgraph in Figure 3.5. An additional seven questions were included to mitigate any recency effects and avoid introducing fatigue. Responses were recorded for each question on a four-point Likert scale using answer options of very poor answer, somewhat poor answer, somewhat good answer, very good answer.

I hypothesize that when subjects read a story statement that introduces a reconsidered intention, subjects will be compelled to think about a story character’s current and future intentions. This activity requires recalling story facts and reasoning over them, resulting in a mental model represented by the QKS intention revision subgraph (Section 3.3). My experiment is designed to evaluate whether the mental model created from reading the exceptional action is accurately represented by the subgraph.
Figure 3.9 Local goal hierarchy change and QKS intention revision subgraph for *Prison Sentence*
3.4.4 Results

Prior to computing inferential statistical analyses, data were inspected and coded to identify missing values, to identify outliers and to ensure that the data were normally distributed. No extreme values were identified in this analysis. My research question was broken down into the three hypotheses that are discussed in Section 3.4.1. Each of the three hypotheses were analyzed using a chi-square test for independence.

Riedl and Young [Rie03] performed a similar experiment to evaluate the intentional structure of IPOCL plans using the QUEST cognitive model. A four-point Likert questionnaire asked subjects whether question-answer pairs from the QKS were good answers to Why? questions. Number values were assigned to each of the four points, and scores were averaged across questions and then an independent t-test was performed. In contrast to this study, the analysis presented below has been adjusted to reflect the new perspectives on Likert scales, namely that contingency tables should be constructed so that chi-squared tests can be used to detect changes in observed responses.

H1. Subjects who do not read the exceptional action statement will rate the Q1 question-answer pair with a low GOA.

_**Prison Sentence.**_ Experimental group responses to Q1 for Prison Sentence are summarized in the contingency tables in Tables 3.6. When I performed the chi-square test for independence, I
Figure 3.11 Local goal hierarchy change and QKS intention revision subgraph for *Farm Boy*
computed a chi-square statistic of 23.33 with a p-value of 0.00 (final column in Table 3.8). This indicates that the experimental groups were drawn from two different populations. Upon further investigation, the Good/Somewhat Good responses of the Control and T1 groups are similar, as are the Bad/Somewhat Bad responses for T2 and T3, indicating agreement between the predicted and observed responses. These results confirm H1 for the Prison Sentence story and indicate that subjects believe the exceptional action has the consequence of Smith’s exoneration goal being unachievable.

Farm Boy. Experimental group responses to Q1 for Farm Boy are summarized in the contingency tables in Tables 3.7. When I performed the chi-square test for independence, I computed a chi-square statistic of 41.3 with a p-value of 0.00 (final column in Table 3.9). This indicates that the experimental groups were drawn from two different populations. Upon further investigation, the Good/Somewhat Good responses of the Control and T1 groups are similar, as are the Bad/Somewhat Bad responses for T2 and T3, indicating agreement between the predicted and observed responses of the question-answer pair. These results confirm H1 for the Farm Boy Sentence story and indicate that subjects believe the exceptional action has the consequence of Hank’s goal of selling his crops being unachievable.

H2. Subjects that do not observe the exceptional action will rate the Q2 question-answer pair with a low GOA.

Prison Sentence. Experimental group responses to Q2 for Prison Sentence are summarized in the contingency tables in Tables 3.6. When I performed the chi-square test for independence, I computed a chi-square statistic of 23.33 with a p-value of 0.00 (final column in Table 3.8). This indicates that the experimental groups were drawn from two different populations. Upon further investigation, the Good/Somewhat Good responses of the Control and T1 groups are similar, as are the Bad/Somewhat Bad responses for T2 and T3, indicating agreement between the predicted and observed responses. These results confirm H2 for the Prison Sentence story and that subjects believe the exceptional action initiates Smith adopting his escape goal.

Farm Boy. Experimental group responses to Q2 for Farm Boy are summarized in the contingency tables in Tables 3.7. When I performed the chi-square test for independence, I computed a chi-square statistic of 41.3 with a p-value of 0.00 (final column in Table 3.9). This indicates that the experimental groups were drawn from two different populations. Upon further investigation, the Good/Somewhat Good responses of the Control and T1 groups are similar, as are the Bad/Somewhat Bad responses for T2 and T3, indicating agreement between the predicted and observed responses. These results confirm H2 for the Farm Boy story and that subjects believe the exceptional action initiates Hank adopting his network destruction goal.
H3. All subjects will rate the Q3 question-answer pair with a low GOA.

*Prison Sentence.* Experimental group responses to Q3 for Prison Sentence are summarized in the contingency tables in Tables 3.6. When I performed the chi-square test for independence, I computed a chi-square statistic of 7.80 with a p-value of 0.55 (final column in Table 3.8). This indicates that the experimental groups were drawn from the same population. The Bad/Somewhat Bad responses of the Control, T1, T2, T3 groups are similar, indicating agreement between the predicted and observed responses. These results confirm H3 for the Prison Sentence story and that subjects organize the *escape* and *exonerate* goals into separate goal hierarchies (*e.g.* *escape* is not a sub-goal of *exonerate*).

*Farm Boy.* Experimental group responses to Q3 for Farm Boy are summarized in the contingency tables in Tables 3.7. When I performed the chi-square test for independence, I computed a chi-square statistic of 2.99 with a p-value of 0.96 (final column in Table 3.8). This indicates that the experimental groups were drawn from the same population. The Bad/Somewhat Bad responses of the Control, T1, T2, T3 groups are similar, indicating agreement between the predicted and observed responses. These results confirm H3 for the Farm Boy story and that subjects organize the *escape* and *exonerate* goals into separate goal hierarchies (*e.g.* *sell crops* is not a sub-goal of *destroy network*).

My three hypotheses have been supported across both narrative texts, allowing me to answer my research question:

**R1. How does an exceptional action affect the QKS intention revision subgraph?** Specifically, the QKS intention revision subgraph is an accurate representation of a reader’s mental model after reading an exceptional action in the narrative texts presented.

### 3.5 Discussion

Agents who adapt and respond to changes in their environment has long been a goal of the artificial intelligence community. This goal is particularly applicable for intentional agents within an interactive narrative, as they need to respond to unexpected user actions. Revising an agent’s intentions in a principled manner that aligns with a model of human comprehension would extend an agent's behavior range in a believable way.

I have defined a model of plan-based intention revision with two key properties. The first is when an agent should *reconsider* an intention. I use a causal link threat to operationalize the *reconsider* function from the BDI control loop. Once an intention is being reconsidered, a second property of intention revision describes how an agent *deliberates* to revise their intentions. Based on when a P-GOAL should be dropped, an agent will drop an intention when no subplan exists for a reconsidered intention. The agent will then replace the reconsidered intention with an achievable intention that is not a subgoal of the
### Table 3.6 Contingency results for Prison Sentence questions

<table>
<thead>
<tr>
<th>Question</th>
<th>Group</th>
<th>Very Bad Answer</th>
<th>Somewhat Bad Answer</th>
<th>Somewhat Good Answer</th>
<th>Very Good Answer</th>
<th>Row Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>Control</td>
<td>0</td>
<td>1</td>
<td>8</td>
<td>16</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>T1</td>
<td>1</td>
<td>1</td>
<td>9</td>
<td>13</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>T2</td>
<td>17</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>T3</td>
<td>15</td>
<td>6</td>
<td>3</td>
<td>0</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>Col. Total</td>
<td>33</td>
<td>13</td>
<td>20</td>
<td>30</td>
<td>96</td>
</tr>
<tr>
<td>Q2</td>
<td>Control</td>
<td>1</td>
<td>0</td>
<td>10</td>
<td>14</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>T1</td>
<td>0</td>
<td>2</td>
<td>8</td>
<td>14</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>T2</td>
<td>16</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>T3</td>
<td>14</td>
<td>7</td>
<td>1</td>
<td>2</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>Col. Total</td>
<td>31</td>
<td>14</td>
<td>20</td>
<td>31</td>
<td>96</td>
</tr>
<tr>
<td>Q3</td>
<td>Control</td>
<td>15</td>
<td>8</td>
<td>1</td>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>T1</td>
<td>12</td>
<td>7</td>
<td>2</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>T2</td>
<td>13</td>
<td>8</td>
<td>1</td>
<td>1</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>T3</td>
<td>13</td>
<td>9</td>
<td>2</td>
<td>0</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>Col. Total</td>
<td>53</td>
<td>32</td>
<td>6</td>
<td>5</td>
<td>96</td>
</tr>
</tbody>
</table>

### Table 3.7 Contingency results for Farm Boy questions

<table>
<thead>
<tr>
<th>Question</th>
<th>Group</th>
<th>Very Bad Answer</th>
<th>Somewhat Bad Answer</th>
<th>Somewhat Good Answer</th>
<th>Very Good Answer</th>
<th>Row Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>Control</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>14</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>T1</td>
<td>1</td>
<td>0</td>
<td>7</td>
<td>18</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>T2</td>
<td>15</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>T3</td>
<td>17</td>
<td>6</td>
<td>1</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>Col. Total</td>
<td>34</td>
<td>14</td>
<td>15</td>
<td>35</td>
<td>98</td>
</tr>
<tr>
<td>Q2</td>
<td>Control</td>
<td>0</td>
<td>1</td>
<td>6</td>
<td>16</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>T1</td>
<td>2</td>
<td>1</td>
<td>6</td>
<td>17</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>T2</td>
<td>16</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>T3</td>
<td>16</td>
<td>7</td>
<td>1</td>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>Col. total</td>
<td>34</td>
<td>13</td>
<td>15</td>
<td>36</td>
<td>98</td>
</tr>
<tr>
<td>Q3</td>
<td>Control</td>
<td>14</td>
<td>7</td>
<td>1</td>
<td>1</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>T1</td>
<td>16</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>T2</td>
<td>17</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>T3</td>
<td>16</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>Col.total</td>
<td>63</td>
<td>24</td>
<td>5</td>
<td>6</td>
<td>98</td>
</tr>
</tbody>
</table>
Table 3.8 QKS intention revision subgraph question results for Prison Sentence

<table>
<thead>
<tr>
<th>#</th>
<th>Pair</th>
<th>EG</th>
<th>Question-Answer</th>
<th>QUEST GOA</th>
<th>chi-square (p-value)</th>
</tr>
</thead>
</table>
| Q1 | E1-S1 | C  | Q: What was a consequence of the warden denying Smith’s new trial request?  
   |       |    | A: That Smith believed exoneration was unachievable.                          | Good      | 41.27 (0.00)         |
|    |       |    | T1 Q: What was a consequence of the warden denying Smith’s new trial request? 
   |       |    | A: That Smith believed exoneration was unachievable.                          | Good      |                      |
|    |       |    | T2 Q: What was a consequence of the Smith sketching and drawing to pass the time?  
   |       |    | A: That Smith believed exoneration was unachievable.                          | Bad       |                      |
|    |       |    | T3 Q: What was a consequence of Smith requesting a new trial?  
   |       |    | A: That Smith believed exoneration was unachievable.                          | Bad       |                      |
| Q2 | G2-E1 | C  | Q: Why did Smith want to escape to Canada?  
   |       |    | A: Because the warden denied his request for a new trial.                      | Good      | 23.33 (0.00)         |
|    |       |    | T1 Q: Why did Smith want to escape to Canada?  
   |       |    | A: Because the warden denied his request for a new trial.                      | Good      |                      |
|    |       |    | T2 Q: Why did Smith want to escape to Canada?  
   |       |    | A: Because Smith passed time by sketching and drawing.                         | Bad       |                      |
|    |       |    | T3 Q: Why did Smith want to escape to Canada?  
   |       |    | A: Because Smith requested a new trial.                                        | Bad       |                      |
| Q3 | G2-G1 | C  | Q: Why did Smith want to escape to Canada?  
   |       |    | A: Because Smith wanted to be exonerated.                                      | Bad       | 7.80 (0.55)          |
|    |       |    | T1 Q: Why did Smith want to escape to Canada?  
   |       |    | A: Because Smith wanted to be exonerated.                                      | Bad       |                      |
|    |       |    | T2 Q: Why did Smith want to escape to Canada?  
   |       |    | A: Because Smith wanted to be exonerated.                                      | Bad       |                      |
|    |       |    | T3 Q: Why did Smith want to escape to Canada?  
   |       |    | A: Because Smith wanted to be exonerated.                                      | Bad       |                      |
Table 3.9 QKS intention revision subgraph question results for Farm Boy

<table>
<thead>
<tr>
<th>#</th>
<th>Pair</th>
<th>EG</th>
<th>Question-Answer</th>
<th>QUEST Predicted GOA</th>
<th>chi-square (p-value)</th>
</tr>
</thead>
</table>
| Q1 | E1-S1  | C  | Q: What was a consequence of the soldiers pillaging Hank’s farm?  
A: Hank believed selling his crops was unachievable. | Good                | 31.06 (0.00)         |
|    |        |    |                                                                                 |                       |                      |
|    |        | T1 | Q: What was a consequence of the soldiers pillaging Hank’s farm?  
A: Hank believed selling his crops was unachievable. | Good                |                      |
|    |        |    |                                                                                 |                       |                      |
|    |        | T2 | Q: What was a consequence of Hank asking for a tax receipt?  
A: Hank believed selling his crops was unachievable. | Bad                 |                      |
|    |        |    |                                                                                 |                       |                      |
|    |        | T3 | Q: What was a consequence of Hank watching cat videos online?  
A: Hank believed selling his crops was unachievable. | Bad                 |                      |
| Q2 | G2-E1  | C  | Q: Why did Hank want to destroy the network?  
A: Because the soldiers pillaged Hank’s farm. | Good                | 21.31 (0.00)         |
|    |        |    |                                                                                 |                       |                      |
|    |        | T1 | Q: Why did Hank want to destroy the network?  
A: Because the soldiers pillaged Hank’s farm. | Good                |                      |
|    |        |    |                                                                                 |                       |                      |
|    |        | T2 | Q: Why did Hank want to destroy the network?  
A: Because Hank asked for a tax receipt. | Bad                 |                      |
|    |        |    |                                                                                 |                       |                      |
|    |        | T3 | Q: Why did Hank want to destroy the network?  
A: Because Hank watched cat videos online. | Bad                 |                      |
| Q3 | G2-G1  | C  | Q: Why did Hank want to destroy the network?  
A: Because Hank wanted to sell his crops. | Bad                 | 2.99 (0.96)          |
|    |        |    |                                                                                 |                       |                      |
|    |        | T1 | Q: Why did Hank want to destroy the network?  
A: Because Hank wanted to sell his crops. | Bad                 |                      |
|    |        |    |                                                                                 |                       |                      |
|    |        | T2 | Q: Why did Hank want to destroy the network?  
A: Because Hank wanted to sell his crops. | Bad                 |                      |
|    |        |    |                                                                                 |                       |                      |
|    |        | T3 | Q: Why did Hank want to destroy the network?  
A: Because Hank wanted to sell his crops. | Bad                 |                      |

48
reconsidered intention. Together these two properties define intention revision for plan-based intentional agents.

To evaluate the effect on human comprehension, I defined a QUEST Knowledge Structure (QKS) intention revision subgraph. Previous QUEST experiments neglected to capture intention revision as part of their evaluations, resulting in a gap in QKS representation. By aggregating existing QKS elements together as a subgraph, I can use the QUEST question-answer protocol to validate the knowledge structures created by the intention revision model and overcome the representation limitation. My evaluation showed that when I introduce the intention revision model in a narrative text, subjects create a mental model that is represented by the QKS intention revision subgraph. With this principled method for intentional agents to revise their intentions, I have evidence to believe that a change in non-player character behavior will make sense to a user.

While this model supports human comprehension of intention revision among agents, some key technical challenges remain. When deliberating, an agent needs to both determine that the reconsidered intention cannot be achieved and then identify a new intention that is not a subgoal. These both require analyzing a planning problem’s solution space, and a typical approach would be to create a plan library using diverse planning and search it for a plan that contains the appropriate intention frame set. However, creating a diverse plan library is computationally expensive and is not guaranteed to explore the solution space in a diverse manner. In Chapter 4 I address this technical challenge by introducing a pre-planning step to identify unique and promising solution areas of the search space.
Planning’s generative nature is an attractive feature for interactive narrative as it decouples the problem from its solution. Since the input is a declarative representation, whenever the situation calls for a new story branch, i.e. a new planning problem, we simply request a new solution with the current state. The representation of a problem allows it to change frequently and easily create story branches without the need to rewrite solver code, search procedures, or represent new knowledge. In my case, when I encounter an exceptional action and the current plan is certain to fail, I desire a story branch that fits my model of intention revision.

To support solution generation in response to problem changes, a number of problem formulation methods have emerged. Perhaps the most popular area of planning research is developing problem formulation methods to improve a planner’s performance. Over the years, the target of these improvements have focused on the heuristic function of a planner, as it has perhaps the biggest impact on a planner’s ability to find solutions. The heuristic function characterizes the amount of work required to reach the goal conditions of the planning problem, in advance of planning and generating solutions. This approach has led to dramatic performance improvements, as the characterization of the problem structure reduces the search space and provides direction for a planner to pursue. Since planning is a P-SPACE complete computation [Byl94], the computation overhead of problem formulation methods done in advance of the planning process is often easily justified through the time they save the planning algorithm.
While the decoupling of problem and solution is a unique advantage planning offers to flexible problem solving, it comes at the cost of disconnecting users from the solutions. Often this is not an issue, as a user is simply interested in obtaining solutions quickly or obtaining a solution with an optimal quality—hence the impact of more robust characterizations for the heuristic function. The disconnection between users and solutions only becomes problematic when the domain representation does not entirely capture the knowledge of the environment. In my case, the deliberation steps for intention revision and the mental state of the IN user are not accounted for. I simultaneously want to both determine if the current planning problem’s solution space contains the agent’s reconsidered intention and identify revision options with specific requirements (e.g. a subgoal). Often [e.g. AM96; Tat95; ML99; Mye02] this gap in representation is resolved by generating a diverse plan set comprised of qualitatively different solution plans to the planning problem, with a user choosing from a set of options.

However, generating a diverse plan set has two major limitations. The first is the inefficient generation of plans. Typical diverse planning approaches require that the user specify to the planner that it must generate $k$ plans to make up the diverse plan set, in hopes the planner finds qualitatively different plans in the $k$. Irrespective of how diverse the planning problem’s solution space is, the planner must generate the $k$ plans it was tasked with. A planning problem with low solution space diversity will force a planner to search fruitlessly for plans with qualitative differences when in fact none may exist. A second limitation is that even with all computation resources allocated to generating $k$ plans, the diversity of the resulting plan set may not be representative of the planning problem’s solution space diversity. Even when a problem has a diverse solution space, say with $k<i$ qualitatively different plans, there is no guarantee that any number of the $i$ plans will be present in the $k$ plans. This limits the comparisons between diverse planners to the plan sets they generate themselves, rather than to the diversity of a planning problem’s solution space, as there is no robust characterization.

A planning problem’s solution space diversity is, in effect, a proxy for the story branches embedded in the planning problem. Diverse planning’s two limitations are particularly problematic for using intention revision as a story branching mechanism. The high computation cost of generating $k$ plans can make a real-time system intractable. The lack of coverage of a planning problem’s solution space diminishes the utility of the $k$ plans, as solutions that conform to my intention revision requirements could be overlooked.

I address the limitations of diverse planning with an approach that exploits an auspicious relationship between the constraints that the IPOCL representation imposes on solutions and the solution space diversity of intentional planning problems. The IPOCL representation requires that every step of a solution plan also be part of an agent’s subplan. The key property of subplans is that they often depend on other agents’ subplans to accomplish their intentions. These subplan dependencies fundamentally shape the solution space diversity of a planning problem and can be computed in advance of plan generation. I devise a problem formulation methodology based on intention dependencies, the intention dependency graph (IDG), to characterize the solution space diversity of a planning problem as qualitatively different partitions. Each partition is comprised of interdependent agent goals that achieve the problem’s goal.
conditions when constructing a planning graph.

I construct the IDG in advance of planning by extending an existing problem formulation structure, story planning graphs, to maintain what agent subplans each action is supporting. Once the story planning graph is complete, I have a set of qualitatively different agent-goal combinations, problem partitions, that I use to direct the planner to qualitatively different solutions. This additional information about the problem structure allows us to be efficient in generating a plan set without the need to specify that a planner generate a $k$ sized plan set at runtime, addressing the first limitation of diverse planning. Additionally, the IDG derived plan set will include all agent-goal combinations that are causally-necessary to solve the planning problem. This enables absolute comparisons between diverse planners as the plan set is reflective of the solution space diversity of the planning problem. What's more, because the partitions are uniquely identified by the agent goals they contain, they support the deliberation process in my intention revision model.

In my evaluation, I use the IDG problem formulation method in conjunction with an existing intentional planner to generate a diverse plan set and compare it to diverse planner results. Using intra plan set diversity measures, I show that using the IDG finds a more diverse plan set using fewer resources, without the need to specify $k$ plans. Additionally, using inter plan set comparisons I show that the IDG plan set subsumes the diverse planner plan sets, demonstrating that, at least on the benchmark problems, the IDG characterizes a planning problem’s solution space diversity of causally-necessary agent-goals. These results give empirical evidence that my approach can address the typical limitations of diverse planning and operationalizes my intention revision model at a technical level, ensuring that exceptional actions do not prematurely end a storyline.

4.1 Intention Dependency Graph

Recall from Section 3.1 that intentional plans add additional constraints to classical plans. They require that each plan step be a part of an agent’s subplan to achieve their goal. A subplan is a kind of relaxed plan where a minimum of one precondition of each action must be satisfied by the agent’s own actions, creating a causal chain of actions. This constraint ensures that intentional plans have believable agent behavior in line with cognitive models of comprehension, namely QUEST [Gra91].

The structure of agent subplans has been leveraged before to increase planner performance with great success in the Glaive planner [WY14]. In this section, I again analyze the structure of agent subplans but this time for insights into how they shape the diversity of a planning problem’s solution space. I extend two of Glaive’s problem formulation structures with a third structure, the intention dependency graph (IDG), to capture the dependencies between agent subplans. From these dependencies I create problem partitions that I use to characterize the diversity of a planning problem’s solution space. To increase the solution space diversity of the Prison example, I add an additional story branch. In response to the DenyTrial user action, Smith may now choose to put in his time and wait to be paroled. Smith’s subplan
to achieve \( \text{paroled}(\text{Smith}) \) is represented as \( \pi'' \) in Figure 4.1.

The first structure Glaive developed to obtain a performance increase was a structure called goal graphs. Goal graphs are a type of relaxed plan that enumerate the possible subplans an agent may pursue to reach their goal.

**Definition 26 (Layered graph)** A layered graph is a graph drawing where every node in the graph is assigned an integer, a layer index. Layers are drawn by grouping together all vertices with the same index. This is done vertically for a planning graph.

**Definition 27 (Goal graph)** A goal graph, \( gg \), is a tuple \( \langle \text{agent}, g, G \rangle \) where \( \text{agent} \) is an agent, \( g \) is \( \text{agent} \)'s goal. \( G \) is a directed layered graph where vertices are actions and layer 0 contains vertices that achieve \( g \). Action \( A \) appears in layer \( 0 < i \) iff \( A \) does not appear in a layer \( k \) where \( k < i \), and \( \text{EFF}(A) \) contains an effect \( e \) that is used as a precondition to an action \( B \) at layer \( i - 1 \). An edge from \( A \) to \( B \) denotes \( \text{EFF}(A) \) used in \( \text{PRE}(B) \). The set of goal graphs for \( \Phi \) is \( GG(\Phi) \).

Intuitively, an agent’s goal can be thought of as the root node of a graph. Constructing the goal graph involves connecting a layer of actions the agent could take to achieve the goal. Connected to these actions is another level of actions that satisfy at least one precondition of the goal-achieving actions. Another layer of actions is added after this and so on until no more actions can be added. Goal graphs for Smith and the warden from the augmented Prison example are presented in Figure 4.2.

Goal graphs are constructed once for a planning problem (line 1 in Alg. 3) and they enable an efficiency gain when constructing story planning graphs by limiting the actions added to a story planning graph to those that are potentially motivated.
Figure 4.2 Goal graphs for Prison planning problem
Figure 4.3 SPG for Prison planning problem
Definition 28 (Story planning graph (SPG)) The SPG of a story planning problem $\Phi$, $SPG(\Phi)$, is a directed layered graph alternating literal $P_i$ and action $A_i$ layers that contain literal and action vertices respectively, written $(P_0, A_1, P_1, ..., A_k, P_k)$ for a $k$-layer graph. Layer $P_0$ contains the literals from $I(\Phi)$ and layers are added until two successive literal layers are equal. Effect edges connect actions to effect literal vertices. Precondition, sub-plan and motivation edges connect literal vertices to actions in the next layer. Potentially motivated actions are added as action vertices in each action layer of the SPG. Literals are added as literal vertices in literal layers from an action’s effects.

The SPG of the Prison example is shown in Figure 4.3. The layer $A_1$ contains the first layer of action vertices, where all actions are applicable and potentially motivated where an applicable action is defined as:

Definition 29 (Applicable action) An applicable action $A$ has all its preconditions satisfied, $PRE(A)$.

Potentially motivated actions are defined as:

Definition 30 (Potentially motivated action) An action $A$ is potentially motivated when $A$ is applicable and is in at least one goal graph, $gg$, with $AGENT(A) = agent(gg)$, and the goal, $g(gg)$, is motivated. $A$’s motivations form a set of agent-goal pairs from the goal graphs it is present in, $MOT(A)$. 

Figure 4.4 IDG for Prison planning problem
Figure 4.5 Original Prison planning problem modified into IDG partition problems
Recall from Definition 13 that an agent goal, $g$, is motivated when $\neg g$ is true in the current state.

In $A_1$ of the SPG, Smith’s $\text{MakeFriends}$ and $\text{Embezzle}$ actions are potentially motivated. Their lone preconditions, $\text{inJail}(Smith)$, are satisfied from the initial state in layer $P_0$, making them applicable. They are also part of Smith’s $\text{exonerate}(Smith)$ goal graph in layer 2 (see Figure 4.2) and the $\text{exonerate}(Smith)$ goal is negated in $P_0$, $\neg \text{exonerate}(Smith)$, so the goal is motivated.

I denote the first potentially motivated action of a sub-plan with a motivation dependency, represented by the dotted lines from the literal $\neg \text{exonerate}(Smith)$ in $P_0$ to the actions $\text{MakeFriends}$ in $A_1$ of the SPG. An action can have multiple motivations if it is present on multiple goal graphs and the motivations are referenced by MOT($A$) (see Definition 30).

Definition 31 (Motivation dependency) A motivation dependency is an SPG edge $s \xrightarrow{m} u$ where $s$ has effect $\neg m$ and $u$ is the first action of a sub-plan in a goal graph $gg$ with goal $m$, $g(gg) = m$.

Both actions in $A_1$ also have preconditions satisfied in the initial state, $P_0$. I denote these with solid edges from $\text{inJail}(Smith)$ and $\text{has(warden,money)}$ to $\text{MakeFriends}$ and $\text{Embezzle}$ in $A_1$, respectively.

Definition 32 (Precondition dependency) A precondition dependency is an SPG edge $s \xrightarrow{p} u$ where $s$ has effect $p$ and $u$ has a precondition $p$. Additionally, the edge $s \xrightarrow{p} u$ must exist in an agent goal graph or be from/to the start/goal step, respectively.

The actions $\text{MakeFriends}$ and $\text{Embezzle}$ in $A_1$ have the effects $\text{friends}(Smith,\text{ warden})$ and $\text{indebted}(Smith,\text{ warden})$ in $P_1$, to which they are connected by effect edges.

Definition 33 (Effect edge) An effect edge is an SPG edge $s \rightarrow p$ where $s$ has effect $p$.

A key to intention frames is that an agent’s sub-plan ($T$) is a kind of relaxed plan where a minimum of one precondition from each action must be satisfied by the agent’s own actions. These satisfied preconditions are represented by precondition edges in the SPG. The remaining open preconditions are satisfied by actions in other agents’ sub-plans. This is because every action in an intentional plan is part of some agent’s sub-plan. Recall that goal graphs enumerate agent sub-plans. The dotted arrowed lines in Figure 4.2 indicate open preconditions of actions that must be satisfied by an action in another sub-plan before being added to the SPG. For instance, in layer 0 in Smith’s $\text{exonerate}(Smith)$ goal graph, the $\text{Testify}$ action requires that $\text{hasTrial}(Smith)$ be true. This can only be accomplished through the warden executing the sub-plan to achieve $\text{hasTrial}(Smith)$ in Figure 4.2. This introduces the final type of edge in the SPG, a sub-plan dependency.

Definition 34 (Sub-plan dependency) A sub-plan dependency is an SPG edge $s \xrightarrow{p} u$ where $u$ is an action in a goal graph $gg$, such that $s$ is not an action vertex in $gg$.

A sub-plan dependency is identified using the dashed lines in Figure 4.4. Note the aforementioned sub-plan dependency between Smith’s $\text{exonerate}(Smith)$ and the warden’s $\text{isDenied}(Smith)$ goals from the $\text{isDenied}(Smith)$ literal in $P_3$ to the $\text{Testify}$ action in $A_4$. 

58
Identifying sub-plan, precondition, and motivation edges while constructing the SPG is important, as they fundamentally shape the solution space of a planning problem. However, these dependencies do not explicitly capture the agent-goal dependencies within the SPG and thus fall short in characterizing the solution space. My contribution—the intention dependency graph (IDG)—addresses this limitation. The IDG identifies the underlying agent goal dependencies to characterize the solution space diversity as partitions of interdependent agent goals. The partitions can then be used to address the limitations of typical diverse planning approaches for intentional planning.

**Definition 35 (Interdependent agent goals)** Interdependent agent goals are a tuple of two agent-goals, \( \langle ag_1, ag_2 \rangle \) where \( ag_1, ag_2 \) are both agent-goal pairs such that there exists at least one subplan or motivation dependency between the goals.

While I could include more representational information in the definition of interdependent agent goals, such as unique subplan IDs or directional dependencies, it would greatly increase the number of interdependent agent goals. When I discuss how the IDG is constructed and when I analyze the space complexity in Section 4.3.1, I will demonstrate how a more specific dependency would greatly increase the complexity of building the IDG.

**Definition 36 (Intention dependency graph (IDG))** The intention dependency graph of a planning problem, \( IDG(\Phi) \), is a directed layered graph represented by the three-tuple \( \langle V, E, \cup_{IDG} \rangle \) where \( V \) is a set of \( n \) vertices \( v_0, v_1, \ldots, v_{n-2}, v_g \) that are each assigned a label, \( label(v_i) \), that consists of a set of interdependent agent goals. The vertex \( v_0 \) represents the initial state vertex and \( v_g \) the goal vertex, both of which have blank labels. \( E \) is a set of directed edges between vertices of \( V \), and \( \cup_{IDG} \) is a vertex labeling function that takes a set of IDG vertices as input and outputs the union of their label contents.

An IDG vertex label is a set of agent goals that are interdependent, either through the presence of sub-plan dependencies or motivation dependencies between the agent goals. Edges between IDG vertices indicate that the successor vertex’s label is the union of its predecessors’ labels (using the labeling function \( \cup_{IDG} \) in Definition 36). A vertex \( v_s \) that connects to \( v_g \) indicates that the \( L(v_s) \) is a set of interdependent agent goals whose subplans are necessary to achieve the literals in the goal condition in the SPG. I refer to \( v_s \) as a solution vertex. I discuss how to translate \( v_s \) into a plan and some of the limitations in Section 4.2.2.

To address the SPG’s shortcomings of not keeping an explicit record of the agent goals needed to add the action to the SPG, I assign each literal and action vertex a *history*. The history is a set of IDG vertices, with each vertex indicating the causally-necessary agent goals needed to achieve the literal or add the action.

**Definition 37 (Causally-necessary agent-goal)** A causally-necessary agent-goal, \( ag \), has a subplan that contains at least one action that has the goal action as a causal descendant.
Algorithm 3 \textit{buildProblemFormulationGraphs}(\Phi)

\textbf{Input:} \( \Phi \) is a planning problem

\textbf{Output:} An IDG for \( \Phi \)

\begin{enumerate}
\item \( \text{GG = } \text{GG}(\Phi) \) \hspace{1cm} \triangleright \text{compute goal graphs once}
\item \( \text{SPG = new SPG} \) \hspace{1cm} \triangleright \text{create an empty SPG}
\item \( \text{IDG = new IDG} \) \hspace{1cm} \triangleright \text{create an empty IDG}
\item \textbf{while} \( P_k(\text{SPG}) \neq P_{k-1}(\text{SPG}) \) \textbf{do}
\item \textbf{for all} ground action \( A \in \Lambda(\Phi) \) \textbf{do}
\item \hspace{1cm} if isApplicable(\text{SPG}, A) \& isMotivated(\text{SPG}, A, \text{GG}) \textbf{then}
\item \hspace{2cm} \text{IDG.setActionPossibleHistory(\text{GG}, A, k)}
\item \hspace{2cm} \text{SPG.addAction(} A, A_k \text{)}
\item \hspace{2cm} \text{HIST(EFF(A)) += HIST(A)} \hspace{1cm} \triangleright \text{effect literal possible history}
\item \hspace{2cm} \text{SPG.addLiterals(EFF(A), A_k)}
\item \hspace{1cm} \text{IDG.setActionPossibleHistory(\text{GG, Goal, k + 1})}
\item \hspace{1cm} \text{SPG.addAction(Goal, A_{k+1})}
\end{enumerate}

\textbf{Definition 38 (Causal parent and child)} A causal parent and child relationship exists when there is a causal link, \( spu \). Step \( s \) is the causal parent of the causal child \( u \).

\textbf{Definition 39 (Causal ancestors and descendents)} The causal ancestors of a step are all steps in the transitive closure of the causal parent relationship. The causal descendents are all steps in the transitive closure of the causal child relationship.

The IDG is constructed in parallel with the SPG, and the crux of IDG construction is how the histories of SPG literals and actions are propagated to the next layer of the SPG.

\textbf{Definition 40 (Literal history)} A literal history is a set of IDG vertices \( \{v_1, v_2, \ldots, v_n\} \) associated with an SPG literal vertex \( \text{HIST}(p) \). Each vertex in \( \text{HIST}(p) \) captures a set of interdependent agent goals that could achieve \( p \). The \( \text{HIST}(p) \) is calculated using the union of histories from actions with effect \( p \).

Before the initial state is added to SPG layer \( P_0 \) (Fig. 4.3), the vertex \( v_0 \) is created with a blank label in \( L_0 \) of the IDG (Fig. 4.4). As each literal is added to SPG \( P_0 \), its history is set to \( v_0 \); see the \( \{v_0\} \) above each literal in \( P_0 \) of the SPG and lines 1–3 of Alg. 4.

In successive SPG layers, potentially motivated actions are added (line 6, Alg. 3) and history is calculated before it is added (lines 4–18, Alg. 4). As in a typical planning graph, once actions and literals are added to the SPG, they are never removed; only the histories are updated. The goal condition of the planning problem, \( \neg \text{inJail(} Smith) \), is added in \( P_4 \) with a history of \( v_3, v_5 \) and \( v_6 \). These three vertices represent the agent-goals necessary to solve the planning problem, and characterize the solution space diversity of the solution plans to the Prison example (\( \pi, \pi^{\prime}, \pi^{\prime}\prime, \pi^{\prime\prime}\prime \)).
**Definition 41 (Action history)** An action history is a set of IDG vertices \( \{v_1, v_2, \ldots, v_n\} \) associated with an SPG action, \( \text{HIST}(A) \). The \( \text{HIST}(A) \) is calculated using the unique combinations of IDG vertices from \( \text{HIST}(\text{PRE}(A)) \) that make \( A \) applicable and \( \text{HIST}(\text{MOT}(A)) \) that make \( A \) motivated.

As action histories are calculated, new IDG vertices are created in two cases. The first is when a new motivation dependency is created in the SPG. In the SPG at layer \( A_1 \), the \textit{MakeFriends} action becomes potentially motivated. Its lone precondition is in \( P_0 \) (solid line from \textit{inJail(Smith)}) as is its motivation (dashed line from \textit{¬exonerated(Smith)}). However, the union of precondition and motivation possible histories (which is \( \{v_0\} \)) does not contain Smith’s goal of \textit{exonerate(Smith)}. Since this does not contain an agent-goal subplan that the action is part of, this means there is a new motivational dependency, so a new IDG vertex, \( v_1 \), is created with a label consisting of the agent-goal pair \( \text{Smith, exonerate(Smith)} \) at \( L_1 \) of the IDG. With this new agent goal accounted for, the \( \text{HIST}(\text{MakeFriends}) \) is set to \( \{v_1\} \) as \text{MakeFriends} is added to the SPG at layer \( A_1 \). The same reasoning of creating a new vertex is applied for \textit{ApproveTrial}, \textit{DenyTrial}, and \textit{Escape} in layers \( A_3 \) and \( A_4 \), captured in lines 9–12 in Alg. 4. It should be noted, though, that these also account for new sub-plan dependencies, which I discuss next.

A second case for a new IDG vertex is when a new sub-plan dependency is introduced in the SPG. In SPG layer \( A_4 \), the \textit{Parole} action is in Smith’s \textit{paroled(Smith)} goal graph and depends on the \textit{deniedTrial} literal in \( P_3 \) for one of its preconditions. However, \textit{Parole} does not contain \( v_3 \)’s labels in its other preconditions; \textit{isGood(Smith)} only has a history of \textit{Smith, paroled(Smith)} in \( v_2 \). For this new sub-plan dependency, a new vertex, \( v_6 \), is created with a label from the union of \textit{label}(\( v_2 \)) and \textit{label}(\( v_4 \)) (lines 13–16, Alg. 4).

However, not all sub-plan dependencies lead to new IDG vertices. In SPG layer \( A_4 \), the \textit{Testify} action depends on \textit{hasTrial(Smith)}, a sub-plan dependency to the warden’s \textit{hasTrial} goal and \textit{ApproveTrial} action. The precondition histories (\textit{label}(\( v_2 \))) already contain Smith’s \textit{exonerate(Smith)} goal, so no new IDG vertex is needed. Additionally, since \textit{label}(\( v_2 \)) subsumes \textit{label}(\( v_0 \)) in the precondition’s possible histories, only \( v_2 \) is used as the \( \text{HIST}(\text{Testify}) \) (lines 17–18, Alg. 4).

As alternating layers of actions and literals are added to the SPG, at some point there will no longer be any remaining applicable actions to add and successive literal layers contents will be equal. At this point, I stop building the SPG (line 4 Alg. 3) and check if the goal conditions have been reached. Note that I would stop SPG construction earlier if I wanted optimal solutions; however I desire to characterize a space of solutions that includes non-optimal ones. The \textit{Goal} action’s history is computed like any other action (line 6, Alg. 4), except it only needs to be applicable, as no motivation is required for the dummy start and goal actions. Once the history is computed, each vertex is linked to \( v_g \) and refer to these vertices as solution vertices (lines 19–21). In the \textit{Prison} example, the SPG achieves \textit{¬inJail(Smith)}. I denote this with the \textit{Goal} action in SPG layer \( A_5 \) and \( v_g \) in IDG layer \( L_5 \).

**Definition 42 (Solution vertex)** A solution vertex is an IDG vertex that is a predecessor to the goal vertex, \( v_g \), and represents a history of the SPG’s dummy \textit{Goal} action. The set of IDG solution vertices
Algorithm 4 `setActionPossibleHistory(GG, A, level)`

**Input:** GG is a set of goal graphs, A is an action, level is the level action is added to

**Output:** Sets HIST(A) to a set of IDG vertices at current SPG level

1. if $A = \text{init}$ then
2. create vertex $v_0$ with empty label
3. $\text{HIST}(A) = \{v_0\}$
4. else
5. for all goal graph $gg \in GG$ do
6. if $A \in gg \mid A = \text{Goal}$ then
   // All IDG vertex combinations to make A potentially motivated
7. for all vertex set $V \in \text{HIST(PRE}(A)) + \text{HIST(MOT}(A))$ do
8. $k = |V(\text{IDG})| + 1$
9. if $AG(gg) \notin \cup_{IDG}(V)$ then $\triangleright$ motivation dep.
10. create $v_k$ with label $AG(gg) \cup \cup_{IDG}(V)$
11. add edge from $\forall v \in V$ to $v_k$
12. $\text{HIST}(A) += v_k$
13. else if $\exists v : |\cup_{IDG}(V) - \text{label}(v)| = 0$ then $\triangleright$ sub-plan dep.
14. create $v_k$ with label $\cup_{IDG}(V)$
15. add edge from $\forall v \in V$ to $v_k$
16. $\text{HIST}(A) += v_k$
17. else $\triangleright$ no new dep., use an existing HIST(A) vertex
18. $\text{HIST}(A) += v_i \in V : |\cup_{IDG}(V) - \text{label}(v_i)| = 0$
19. if $A = \text{Goal}$ then $\triangleright$ connect the solution vertices to $v_g$
20. create vertex $v_g$ with empty label
21. add edge from $\forall v \in \text{HIST}(A)$ to $v_g$
is $\text{SOL}(IDG)$. Each solution vertex is a unique set of interdependent agent-goals that achieves the goal conditions in the SPG.

In the IDG of the *Prison* planning problem, $v_g$ is a successor to three other IDG vertices, the solution vertices of the planning problem, $\text{SOL}(IDG)$. The first, $v_3$, captures a partition of the search space that includes the original plan, $\pi$, where the warden (user) approves Smith’s new trial. A second vertex, $v_5$, represents a search space partition for the solution plan where the warden denies Smith’s request and Smith responds by escaping ($\pi'$). Finally, $v_6$ is a search space partition that includes the $\pi''$ plan I introduced in this section. Together these solution vertices partition the search space into areas that contain interdependent agent goals that are causally-necessary to solve the planning problem $\Phi$.

### 4.2 Generating a Diverse Plan Set

My goal with intention dependencies is to address the limitations of diverse planning by characterizing the diversity of a planning problem’s solution space with a problem formulation method before planning starts. From this characterization I can identify a diverse plan set in an efficient manner and be assured it is representative of the planning problem’s solution space. While the IDG identifies interdependent agent-goals that divide the search space into qualitatively different partitions, two challenges remain. The first is to direct a planner to the different partitions. I accomplish this by modifying the initial state of the planning problem for each IDG solution vertex to only include the agent-goals in the vertex’s label. A second challenge is to ensure that all the agent-goals in the modified planning problems are pursued. A slight modification to an intentional planner’s heuristic function addresses this challenge. Together, these two steps direct a planner with the information to generate an exemplar plan for each IDG solution vertex, should one exist. I discuss the cases when the problem partitions represented by a solution vertex do not contain a solution plan in Section 4.2.2.

#### 4.2.1 Plans from IDG Solution Vertices

Recall that I construct the IDG for a planning problem $\Phi$. Within the initial state of $\Phi$ will be the goals each agent desires. These agent goals amount to the options a planner has to structure a solution plan. The IDG solution vertices organize these goals into promising search areas for solutions. I constrain the search a planner performs by modifying the initial state $\Phi$ to reflect the agent goals in each solution vertex.

I continue with the *Prison* example, where I desire to generate exemplar intentional plans for each of the three IDG solution vertices, $v_3, v_5, v_6$. I begin by directing a planner to different IDG partitions by creating a partition problem for each IDG solution vertex.

**Definition 43 (Partition problem)** A partition problem, $\Phi'_n$, is a modified version of a planning problem $\Phi$, whose initial state contains only those agent goals in an IDG solution vertex, $v_n$. The goal conditions,
agents, objects, and action schema of the original planning problem remain the same. The partition problem set of the planning problem contains one partition problem for each solution vertex in \( \text{Sol}(IDG) \).

The initial state of the \textit{Prison} planning problem and the resulting partition problems are detailed in Figure 4.5. Note the difference in agent goals between \( \Phi'_3, \Phi'_5, \Phi'_6 \) and similarity of \( \Phi'_3 - v_3, \Phi'_5 - v_5 \) and \( \Phi'_6 - v_6 \). These partition problems ensure that solutions will only contain agent-goals from the IDG solution vertices.

4.2.2 IDG Advantages and Limitations

The IDG solution vertices represent a characterization of the solution diversity that can be used to address two important limitations of diverse planning. First, rather than specifying a planner to generate \( k \) plans without knowing how many different plans there are in the solution space, each IDG solution vertex can direct a planner to a problem partition that contains a specific agent goal set. This is advantageous to a planner, as without the problem partition it is implicitly considering all possible combinations of agent goals (\( 2^{|ag|} \), where \( ag \) is the number of unique intentions). A planner can then generate a representative exemplar solution plan (or more) from this partition. The solution vertices all differ by at least one agent goal, ensuring that plans generated from them are qualitatively different from each other.

Second, there is no benchmark to assess a diverse planner’s performance with respect to the planning problem’s solution diversity. Since the IDG keeps track of the motivation and sub-plan dependencies between agent goals, its solution vertices characterize the solution diversity of causally-necessary agent-goals that could achieve the goal state. This characterization can serve as a benchmark to assess a diverse planner’s performance relative to the problem, whereas diversity comparisons are only done relative to one another.

The IDG has a limitation that causes it to overestimate the number of solution vertices. The SPG does not use \textit{mutex} edges to capture precedence and interference relationships between actions. Like other plan graphs without mutex edges, it causes unreachable plans to be considered as solutions, resulting in unreachable agent goals and impossible histories of IDG vertices. The overestimation can be calculated from the number of solution vertices that do not generate solution plans.

4.3 Evaluation

The generative nature of planning is a compelling fit for computer systems in uncertain environments. When unforeseen events occur and the problem changes, a planner can simply solve the problem again with the present state. However, this has limitations when humans are involved. In my case, new plans that support intention revision need to account for the existing mental state of the user. I have defined
these features in Chapter 3, but they are not represented in individual plans themselves but rather the solution space of the problem.

My goal with the IDG is to support an agent revising their intentions by both efficiently identifying a diverse plan set (capturing an agent’s options) and ensuring that the plan set characterizes a planning problem’s solution space diversity (the best options are known). First, using intra plan set diversity, I demonstrate that the IDG can direct a planner to a more diverse plan set using far fewer plans than diverse planning. This plan set provides the options for resolving an exceptional user action. Second, I use an inter plan set measure to show that the IDG plan set subsumes plan sets from existing diverse planners. This demonstrates that a plan set generated from the IDG’s partition problem characterizes the original problem’s solution space, ensuring that the options in the plan set are representative of the solution space and not a function of an unfortunately directed planner. The characterization enables more robust comparison between diverse planner performance and a planning problem’s solution space diversity.

The planning community has long been a supporter of benchmark problems [Hof06; Hel06; HD09; Col12]. The community’s most widespread evaluation has been the International Planning Competition (IPC). The IPC benchmark problems are designed to assess planners on some efficient use of resources, be it power, number of steps, etc. Unfortunately, high solution space diversity has not been designed into the benchmark problems. As such, there is little guidance or methods to develop diverse planning domains and problems, further contributing to solution space diversity being an understudied area. Without appropriate benchmarks, comparisons between planners are limited because the efficacy of the search method and the inherent diversity of a problem are confounded. To address this limitation, I designed a problem, Politico, to have high solution diversity. Solutions to Politico, along with the benchmark problems, are used with both intra- and inter-solution diversity to compare the IDG against previous results [FW16]. The Heist problem (no solutions for previous work) and BLP-W/L (ambiguous PDDL) were not used.

### 4.3.1 Time and Space Complexity

Problem formulation methods are afforded latitude when it comes to complexity, owing to the fact that classical planning belongs to the P-SPACE complexity class [Byl94]. The additional time complexity required to create the IDG arises from calculating the history of each SPG vertex. Literal vertices are straightforward, as they are simply the union of the action histories that are connected by effect edges. The union can be performed in the worst case $O\log^* n$ [TL84] where $n$ is the number of effect edges connected to the literal. For each of these

$$C = |\text{HIST}(\text{MOT}(A))| \times \prod_i |\text{HIST}(\text{PRE}_i(A))|.$$  

For each of these
configurations, there is \( |\text{PRE}(A)| \) unions performed for a \( U = |\text{PRE}(A)| \log^* |ag| \) cost where \( ag \) is the largest IDG label. Once the label is determined, the IDG is searched for an existing vertex at cost \( I = n \log n \) where \( n \) is the size of the \( V(IDG) \) at the current level. This is a per-vertex calculation and is done once per-vertex in the SPG, where the number of action vertices is \( A \). In the worst case, the time complexity for computing the history of the action vertices in the SPG is \( O(CUIA) \).

Together, computing action and literal histories adds time complexity to Glaive’s planning graph but still remains in the polynomial complexity class along with the original Graphplan planning graph. The final time complexity of computing the histories in the SPG is the literal histories plus the action histories \( O(\mathcal{L} + CUIA) \).

The space complexity cost of the SPG is maintaining the history of IDG vertices in the SPG vertices. It is possible to have SPG vertices with a history of the entire IDG vertex set. While not every SPG vertex will have such a large history, it is still the worst case and therefore maintaining the histories has a space complexity of \( O(N^2 |ag|) \), where \( N \) is the number of SPG vertices and \( |ag| \) is the largest possible IDG label.

In addition to the time and space complexity costs added to the SPG, the IDG algorithm itself has a space complexity cost to account for. First, the maximum number of vertices in the IDG is a function of the number of agent goals in the planning problem, \( |ag| \). An IDG vertex label could be any unique combination of agent goals, and so there could be at most \( 2^{|ag|} \) IDG vertices.

A second space complexity involved in constructing the IDG is the number of edges between IDG vertices. As actions are added to the SPG, IDG vertex labels may be combined to create new vertices. When these new vertices are created, edges are added to the vertices that are being combined. In the worst case, every new vertex requires an edge for every agent goal in the label. This means that every one of the \( |ag| \) vertices in layer one would be connected to every vertex in which that label is present, representing the highest degree vertices in the IDG. Each of these \( |ag| \) vertices could be connected to as many as \( 2^{|ag|-1} \) vertices, meaning as many as \( 2^{|ag|} \) edges. A worst case IDG with \( 2^{|ag|} \) vertices plus \( 2^{|ag|} \) edges has a space complexity of \( O(2^{|ag|} + 2^{|ag|}) \). I present an example of the worst-case IDG in Figure 4.6.

Overall, constructing the IDG can add an exponential space cost. While this is undesirable upon first glance, there are several factors to consider. First, the exponent is the number of agent goals and is typically a small number. In fact the narrative benchmark problems I discuss in Section 5.3 have at most seven. Second, the dependencies between agent goals is typically not so intertwined as to create a worst-case number of IDG vertices. A smaller number of vertices also limits the number of edges created, further reducing the space required. A third factor is how well the IDG performs against diverse planning. If constructing the IDG generates a more diverse plan set with fewer plans, the total runtime may be less than diverse planning. Outside of these complexity factors, it is worth noting the IDG complexity is driven by the vertex labels. Alternate labels reflecting a character’s unique subplan to accomplish their goal, rather than an agent-goal, could be used to identify a space of subplan combinations to achieve the goal state. This label representation may render computing the IDG intractable, due to the exponential
cost and I leave this insight for future work outside this dissertation.

4.3.2 Designing a Problem with a Diverse Solution Space

In contrast to most benchmark problems that are designed to exercise a planner’s ability to generate a single optimal solution, a problem’s solution space diversity is quite difficult to evaluate. The crux of the issue is that there is no way to compute when a planner has generated enough plans to cover or represent a problem’s solution space. This makes it difficult to design non-trivial benchmarks with a large enough solution space to separate a planner’s ability to generate a diverse solution space and the diversity of a problem’s solution space. As such, there are no diverse problems to adapt from the IPC benchmarks.

Within the interactive narrative field, one might expect intentional planning problems with a diverse solution space. A diverse solution space would support a branching narrative, adapting to user input and creating a rich user experience with the choices afforded by the story branches. It is somewhat surprising, then, that Farrell and Ware’s [FW16] analysis revealed that there is very little diversity in the solution spaces of a set of public intentional planning benchmarks. The lack of diversity is likely due to the focus on augmenting and extending the intentional planning representation.

Despite the friction present between a problem’s solution diversity and diverse planning, I take a first step toward creating a set of planning problems with diverse solution spaces. I developed the Politico
Figure 4.7 Politico problem
problem with a diverse solution space that can be readily computed by hand. Additionally, the diversity is based on different agent goals being pursued across solutions. Politico begins in a fictitious setting where the government is without a designated survivor. A political upstart agent (the user) who desires to become head of state can interact with four (non-player) agents who possess different skills and resources. A user can either take or ask other agents for their skills and resources, and then combine them to become the head of state. However, if the user takes (vs asks) resources from the other agents they will become rebellious, band together, organize their own political machine, and foil the user’s plans by installing their own head of state. This dynamic interaction between agents is an exemplar of how an intentional plan-based IN could function in response to user actions.

The Politico problem is presented in Figure 4.7 and example plans in Figure 4.8. The problem has nine unique combinations of agent goals to reach a solution; however, for each agent goal in the initial state, there is only one possible subplan to achieve it. This ensures a high solution space diversity in a problem that is computable by hand but complex enough that some of the agent-goal combinations will not be found by diverse planners. If I extrapolate from the example plans and construct all the plans that achieve the goal conditions, there are nine unique plans. I summarize the agent-goals that structure the plan contents of the nine unique solutions in Table 4.1.

![Figure 4.8 Politico solution plans for π₁, π₄, π₉](image-url)
### 4.3.3 Efficiently Generating a Diverse Plan Set: Intra Plan Set Diversity

Recall that current diverse planning approaches require that a user specify a planner to generate \( k \) diverse plans, irrespective of if there are actually \( k \) diverse plans in the solution space. Unless a user is an extremely fortunate estimator, the discrepancy between \( k \) and the solution space diversity will lead to a number of redundant and duplicate plans. The IDG is a problem formulation method designed to replace \( k \) with an estimate of the problem’s solution space diversity, informed by analyzing the problem structure for intention dependencies.

Politico and the benchmark problems are used to evaluate whether using IDG is more efficient than previous results [FW16]. The Heist problem (no solutions for previous work) and BLP-W/L (ambiguous PDDL) were not used.

I used the IDG to analyze each problem in Table 4.2 before planning. This analysis identified solution vertices, where each vertex is comprised of a set of interdependent agent goals that were used to generate the problem partitions. I used the Glaive planner to generate a solution to each problem partition, resulting in the IDG\textsubscript{Glaive} problem set. While I used the Glaive planner, any intentional planner could be used, as my goal is only to demonstrate that a diverse plan set can be produced from IDG solution vertices. The results of this process are presented in row 7 of Table 4.2, where the number of solution vertices are in brackets and the number of plans generated is adjacent (1 exemplar plan per solution vertex-partition problem pair). It should be noted that across problems, every solution vertex led to a solution plan. While this is an attractive result, it is a function of the problem structure and does not demonstrate empirically that the IDG overestimates a problem’s solution space diversity that I detail in Section 4.2.2.

#### Table 4.1 Summary of intentions necessary to solve the politico problem

| Plan | \( |I(\pi)| \) | politician (hasPower politician) | junta (hasPower junta) | tinker (isPartisan tinker) | tailor (isPartisan tailor) | spy (isPartisan spy) | tinker (trust tinker) | tailor (trust tailor) | soldier (trust soldier) | spy (trust spy) |
|------|----------------|-------------------------------|------------------------|------------------------|------------------------|------------------------|-------------------|-------------------|------------------------|-------------------|
| \( \pi_1 \) | 3  | X | X | X |
| \( \pi_2 \) | 3  | X | X |
| \( \pi_3 \) | 3  | X | X |
| \( \pi_4 \) | 4  | X | X | X |
| \( \pi_5 \) | 3  | X | X | X |
| \( \pi_6 \) | 3  | X | X | X |
| \( \pi_7 \) | 3  | X | X | X |
| \( \pi_8 \) | 4  | X | X | X | X |
| \( \pi_9 \) | 1  | X |
Table 4.2 plan set diversity of story benchmarks [War14] compared against previous results [FW16].

<table>
<thead>
<tr>
<th>plan set ((\Pi))</th>
<th>Ark</th>
<th>Fantasy</th>
<th>Politico</th>
<th>Space</th>
<th>Western</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(\Pi)</td>
<td>Uni</td>
<td>Div</td>
<td>(\Pi)</td>
<td>Uni</td>
</tr>
<tr>
<td>BFS</td>
<td>96</td>
<td>1</td>
<td>0.00</td>
<td>91</td>
<td>1</td>
</tr>
<tr>
<td>BFS-n2</td>
<td>98</td>
<td>1</td>
<td>0.00</td>
<td>98</td>
<td>1</td>
</tr>
<tr>
<td>Glaive</td>
<td>81</td>
<td>1</td>
<td>0.00</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>Glaive-n2</td>
<td>85</td>
<td>1</td>
<td>0.00</td>
<td>100</td>
<td>2</td>
</tr>
<tr>
<td>IDG(_{Glaive})</td>
<td>(5)</td>
<td>5</td>
<td>0.43</td>
<td>(5)</td>
<td>5</td>
</tr>
</tbody>
</table>

To compare the \(IDG_{Glaive}\) plan set against other planners, I use two intra plan set measures. The first is an intuitive measure designed to count the unique plans of a plan set based on their agent goals:

\[
Uni(\Pi) = \frac{1}{2} \sum_{\pi_1, \pi_2 \in \Pi} \begin{cases} 
0, & ag(\pi_1) = ag(\pi_2) \\
1, & ag(\pi_1) \neq ag(\pi_2)
\end{cases}
\]  

(4.1)

where \(ag(\pi_1)\) and \(ag(\pi_2)\) are the agent goals from each plan’s intention frames (Definition 12) in the plan set, \(\Pi\). The \(Uni(\Pi)\) metric counts plans as different if they differ by at least one agent-goal and provides a simple indication of how many different agent goal combinations are present in the plan set.

The second measure is the typical plan set diversity measure that calculates the average pair-wise distance between all plans,

\[
Div(\Pi) = \frac{\sum_{\pi_1, \pi_2 \in \Pi} \delta(\pi_1, \pi_2)}{|\Pi| \times (|\Pi| - 1)}
\]  

(4.2)

where \(\Pi\) is a plan set, \(\pi_1\) and \(\pi_2\) are plans, and \(\delta\) is a distance metric between two plans. Since my goal is to assess the space of agent goal changes (intention revisions) in a plan set, I define a Jaccard-based, domain-specific distance metric that compares agent goal differences,

\[
\delta_{ig}(\pi_1, \pi_2) = 1 - \frac{|ag(\pi_1) \cap ag(\pi_2)|}{|ag(\pi_1) \cup ag(\pi_2)|}
\]  

(4.3)

where \(ag(\pi_1)\) and \(ag(\pi_2)\) are the agent goals in each plan. In order to capture the solution space properties pertinent to intention revision, namely the deliberation steps, I required a domain-specific metric that captured the semantics (agent-goals) over a domain-independent one.

In Table 4.2, I show the results of the \(Uni(\Pi)\) and \(Div(\Pi)\) measures. I observe that across all five problems, the BFS and Glaive diverse planners’ plan sets (rows 1–4) consist of a number of redundant
plans, as evident by the disparity between $|\Pi|$ and the unique plans $Uni(\Pi)$. This explains the low $Div(\Pi)$ calculations of the Ark, Fantasy, Politico and Western plan sets in comparison to $IDG_{Glaive}$. In contrast, the plans generated with partition problems from the IDG were both fewer and had greater diversity across the same problems.

In my diverse problem, Politico, and the Space problem, several of the diverse planners generate more than a single unique plan ($Uni(\Pi)$). To verify that low diversity scores are not the result of a disparity between the number of plans requested and the number found (e.g. all the unique plans are found in the first few plans), I plot the number of unique plans in a plan set (y-axis) as plans are generated (x-axis) in Figure 4.9 to determine how quickly each planner generates a plan from a new area of the solution space. I observe that $IDG_{Glaive}$ has a unique plan for each solution vertex; note the steep slope in the plot and that $Uni(\Pi)$ and $|\Pi|$ are equal. Interestingly, the diverse Glaive planner appears to generate more unique plans earlier than Glaive-n2, but Glaive-n2 ultimately generates one more unique plan. While these results are limited as they assess the efficacy of two diverse planning methods on a single diverse problem, they demonstrate that the low diversity score is not the result of an unfortunate choice of $k$. Rather, unique plans are generated throughout the plan sets of diverse planners, and a large $k$ is needed to explore the entire solution space diversity.

Finally, results from the Space problem highlight another property of the IDG. Note the greater $Div(\Pi)$ results for the diverse planners on the Space problem over other problems, and that Glaive and Glaive-n2 $Div(\Pi)$ results were both greater than $IDG_{Glaive}$ in the Space problem. This results from diverse planners generating plans with agent goals not causally-necessary to achieving the goal state. Recall, IDG solution vertices contain sets of causally-necessary agent goals. They will not include agent-goals that ‘pad’ solutions with non-intervening actions. While padded agent-goals in solution plans can increase the diversity of a plan set and do characterize another property of a problem’s solution space, I ultimately view it as separate problem that is not addressed by the IDG.

### 4.3.4 Characterizing a Problem’s Solution Space: Inter Plan Set Diversity

A second limitation of diverse planning is that the resulting plan sets do not characterize a problem’s solution space diversity. This leads to diverse planners only being compared relative to one another and not in terms of absolute performance or a problem of known difficulty. The IDG solution vertices are a characterization of the solution space that supports absolute evaluations. I demonstrate its appropriateness as a characterization of solution space diversity with an inter plan set measure,

$$\beta(\Pi, \Pi_{IDG}) = |\Pi' - \Pi'_{IDG}|, \quad (4.4)$$

where the members of $\Pi'_{IDG}$ and $\Pi'$ are sets of agent-goals, one for each plan in a diverse planner’s plan set. $\Pi_{IDG}$ is the solution vertices of the IDG that have been shown to produce plans. The $\beta$ measure assesses the IDG’s subsuming of another plan set. To be considered as a characterization of solution
space diversity, the agent-goal sets of $IDG_{Glaive}$ should subsume agent-goal sets from the four diverse planners.

The IDG as a characterization enables us to differentiate between poor diverse planning and a planning problem with inherently low solution diversity. I calculate the IDG solutions not present in the candidate plan set ($\Pi$) using

$$\Delta(\Pi, \Pi_{IDG}) = |\Pi_{IDG} - \Pi|.$$  \hspace{1cm} (4.5)

In the $\beta$ columns of Table 4.3, the zeroes indicate that the IDG subsumes the diverse planner results for the Ark, Fantasy, Politico, and Western problems. Again, the Space problem is the exception due to plans that contain agent goals not causally-necessary to solve the planning problem. Despite the Space results, I conclude that the IDG is an appropriate characterization of the causally-necessary agent goals in the solution space. The zeroes in the $\Delta$ column of Western show that the low $Div(\Pi)$ across diverse planners is due to the low diversity in the planning problem’s solution space diversity, not poor diverse planning. This example highlights the value of a characterization of a planning problem’s solution diversity.
Table 4.3 IDG as a characterization of solution space diversity

<table>
<thead>
<tr>
<th>Π</th>
<th>Ark</th>
<th>Fantasy</th>
<th>Politico</th>
<th>Space</th>
<th>Western</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>β</td>
<td>Δ</td>
<td>β</td>
<td>Δ</td>
<td>β</td>
</tr>
<tr>
<td>BFS</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>BFS-n2</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Glaive</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Glaive-n2</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

4.4 Discussion

Agents who revise their intentions in response to their environment are an important part of interactive and autonomous systems. An intentional planning problem’s solution space diversity is an essential part of my plan-based model of intention revision, as it informs the deliberation steps an agent makes when revising an intention. To address the limitations of diverse planning that lead to a weak relationship between the solutions a diverse planer generates and the problem’s solution space diversity, I have formalized a problem formulation methodology, the IDG.

Before planning begins, the IDG characterizes the solution space diversity of an intention planning problem by using interdependent agent goals. These interdependent goals are identified through motivation and subplan dependencies of individual agents while the story plan graph is being constructed. Using an intra plan set diversity, I demonstrate that the IDG is more parsimonious with the number of solution plans it generates by comparing it against the diversity of existing diverse planners. Second, I use inter plan set diversity to demonstrate that the IDG can be used to generate solutions that existing diverse planners do not generate and that characterize the problem’s solution space of causally-necessary agent-goals. This makes it appropriate as a benchmark to compare diverse planners against the solution space diversity of a problem. Together, these two measures show that the IDG supports intention revision deliberation and can be useful in ensuring the generation of coherent story branches.
Interactive narrative (IN) leverages our cognitive resources for increased engagement in both learning and entertainment contexts. To endow an IN user with a sense of agency, users are presented with choices and allowed to interact with a virtual environment in ways that shape their experience. When the IN author has specific goals, be it narrative-centric such as an important event, or overall experience quality, an experience manager (EM) is often employed to preserve these goals while enabling user agency.

When a user exercises their agency by taking an action that introduces a causal-link threat to the current plan, an EM will take steps to mediate the current, unplanned state of the story world and the overall authorial goals. This specific type of mediation is called accommodation, as it accommodates the user’s actions by creating a story branch to fit the actions (I discuss mediation in greater detail in Section 2.6). Previous mediation research on accommodation prescribes that story branches should be coherent with the original story. However, the concept of coherence between intentional plan-based story branches has yet to be defined.

The attractive feature of an intentional plan as the basic story representation for INs is that it offers an IN author believable character behavior. Intentional plan representations of narrative have progressed from non-player character agents who always achieve their intention [Rie03], to agents who abandon their intention at the first complication [WY14], and now to agents who can deliberate over current intentions and choose to revise their intentions [ABY18]. This range of agent behavior enables an experience
manager to create a coherent story branch that accommodates exceptional user actions and preserves believable character agent behavior when a story would otherwise end.

Underlying the justification of branching stories in INs is a long-held assumption that creating a story branch to accommodate a user’s actions improves their experience and is therefore a desirable feature for experience management. Evaluations of plan-based narrative have focused on comprehension (e.g. [CR16]) and specific narrative devices (e.g. [Che15]) while development and evaluation of mechanisms for creating story branches has received comparatively less attention [RB14]. Despite the continued research into interactive narrative and the commercial success of branching story games, it is still unclear how branching stories affect user experience. The appeal of interactive narrative relies on our proclivity to relate to stories and our experience of them, therefore it is necessary to investigate how the mechanisms used to create story branches affect user experience.

Investigating the effects associated with branching stories is difficult. Simply having the branching mechanisms, in my case intention revision, to create coherent accommodations is not enough to claim they improve user experience. Tied up in the concepts of coherence and accommodation are three properties to account for when assessing user experience. The first is the type of user interaction. This can range from choice-based, choose-you-own-adventure style interactions to more open-world environments where a user can interact with the story world however they choose. A second property is the story branching mechanism. An advantage to using intentional plans is it creates an expectation of character behavior on the user’s part, as branching mechanisms can employ differing strategies based on the structural differences between the original and new plan. Finally, the third property is the measures used to evaluate the effects of a story branching mechanism in response to user interaction. Creating story branches offers the opportunity for dynamic intentions but also risks introducing plot holes. Measuring how branching mechanism’s affects users allows us to evaluate a branching mechanism’s relationship to preserving experience quality.

To test the assumption that accommodating user actions improves user experience, I have conducted an experiment that accounts for the three aforementioned properties. First, I use narrative texts as the medium. To separate the effects of an interactive narrative and the branching mechanism on user experience, I simulate exceptional user actions in an IN that initiate the creation of a story branch. The resulting set of story branches is then translated to text and read by subjects. Second, I use intention revision as the story branching mechanism. Based on the three intention revision deliberation steps I defined in Section 3.2, I develop a taxonomy of coherent story branch classes. This taxonomy enables us to summarize and compare a user’s experience of multiple story branches within the same story. Finally, I use highly cited and well validated measures to evaluate intention revision as a mechanism for creating story branches. I collected subject responses to the narrative transportation scale (TS, [GB00]) and perceived interest questionnaire (PIQ, [Sch97]) after they read the generated text. When intention revision is used to create a story branch, it leverages our cognitive processes to think divergently about an agent’s future intentions and I hypothesize this process increases narrative transportation and perceived
My results show that when my plan-based formalization of intention revision is used to create story branches for simulated interactive narratives, subjects who read the resulting text-based narratives reported higher narrative transportation and perceived interest than those who read their linear, non-branching counterparts. Additionally, the results also suggest that a more specific type of intention revision (revenge) could increase TS and PIQ scores even more. Overall, these results support the assumption that branching stories can improve experience, and provide a foundation for future work investigating the effects of intention revision in an interactive narrative.

5.1 Intention Revision as a Branching Mechanism

Up until this point, I have discussed intention revision as the only type of a story branch, namely that intention revision is a consequence of resolving a causal-link threat introduced by a user. However, in the deliberation steps of intention revision it is possible to identify plans that do not require an intention revision. These plans present branching options that are not as cognitively onerous as intention revision, as they, at a minimum, do not require a change in local goal hierarchy in the QKS. In the interest of furthering the discussion surrounding accommodating user actions, I formally define these intentional plans in a taxonomy of story branches in Figure 5.1.

While I have referenced the concept of a story branch throughout this document, I formally define it in an intentional planning context here:

**Definition 44 (Story branch)** A story branch, $\psi$, is a tuple $\langle \pi, \chi, \pi', \text{TYPE} \rangle$ where $\pi$ is the original intentional plan that solves the problem $\Phi$, $\chi$ is an exceptional user action that occurs at step $t$ initiating a reconsidered intention in $\pi$, and $\pi'$ is the new intentional plan that solves the problem $\Phi'$. The initial state

![Figure 5.1 A taxonomy of accommodation types to causal-link threats](image-url)
of $\Phi'$ is the execution state after $\chi$ is executed and $G, A, O, \Lambda$ are equal to $G(\Phi), A(\Phi), O(\Phi), \Lambda(\Phi)$. The TYPE is one of four classes of story branches whose labels are DUP, SUBPLAN, SUBGOAL, IR.

Intentional plans are structured to capture goal-oriented behavior of agents. As such, deviating from the original plan and introducing a new plan as a story branch will have implications across agent behaviors and user mental models (such as the QUEST model I discussed in Section 3.3). I capture these implications in two structural differences between the original’s and the new plan’s intentions. The first is the goals agents pursue. These goals are represented by the $g$ element within an intention frame and filter the applicable actions to those that are motivated. While only logical equivalence can be used to compare goal literals, I instead use a similar-different spectrum to account for the semantic models users have attached to the literals. A second difference is the action sequences by which agents achieve their goals. Immediately abandoning a goal after encountering a causal-link threat calls into question how committed the agent really was to the intention. A more resilient approach is to find another action sequence to accomplish the same goal. Again, I use a similar-different spectrum to account for the semantic models users have attached to action sequences. These two structural differences are captured in the taxonomy of accommodations in Figure 5.1.

Within my taxonomy I focus on quadrant 1 (similar goal and subplan), where a first class of story branches are those that perhaps do not even qualify as story branches. It is possible to have multiple plan steps establish the same precondition. However, a planner will only causally link one of the established preconditions. A planner can easily resolve a causal-link threat introduced at runtime by causally linking the second precondition. This requires no actions in the story branch that are not in the original plan. To account for this class of accommodation, albeit a rare one, I define the duplicate achievement class of story branches:

**Definition 45 (Duplicate achievement story branch)** A duplicate achievement story branch, $DUP(\pi, \chi, \pi')$, is where $\pi'$ has a step that resolves the causal-link threat introduced by $\chi$ and the steps in $\pi'$ are subsumed by the steps in $\pi$, $S(\pi') - S(\pi) = 0$.

The duplicate achievement class of story branches has no difference in agent goals or agent sub-plans; see quadrant 1 in Figure 5.1. The only difference is the change in causal-linkages between plan steps due to the causal-link threat. In a slightly modified version of the Prison example, Smith no longer needs the warden’s approval to get a new trial. In this story world, in order to Testify it is simply a requirement that the warden has responded to Smith’s request; however to get his new trial Smith must FilePapers. I can see in the original plan that both the ApproveTrial and FilePapers establish the $\text{hasTrial(Smith)}$ precondition. Therefore, when the warden unexpectedly denies the trial request, Smith still has the FilePapers result to establish $\text{hasTrial(Smith)}$. I indicate this change in causal connections with a bold line in the bottom diagram in Figure 5.2.

A second class of story branches comes from plans that have the same agent goals but different subplans to achieve them. These plans are represented in quadrant 2 in Figure 5.1. An example of such a
Figure 5.2 A modified Prison planning problem $\Phi$ that demonstrates the original plan (top) relative to the structure of a duplicate achievement story branch (bottom), $\text{DUP}(\pi, \chi, \pi')$.

The story branch is in Figure 5.3, where after being denied by the warden, Smith requests a new trial from the warden’s superior, the superintendent. The subplans differ, but Smith’s goal of $\text{exonerate}(\text{Smith})$ remains the same.

**Definition 46 (Subplan Story Branch)** A subplan story branch, $\text{SUBPLAN}(\pi, \chi, \pi')$, is a branch where the active intentions are preserved between the plans $I^\pi(\pi) - I(\pi') = 0$ and the subplan steps of the reconsidered intention frame $T(I^R)$ in $\pi$ do not subsume the new subplan steps for the same intention in $\pi'$.

In contrast to the duplicate achievement, users will be required to integrate the new subplan information in their mental models. However, in the QUEST knowledge structure the new actions will be placed within the same local goal hierarchy. This is not to say that alternative subplans do not have semantic consequences. Outside of QUEST, alternative subplans have been used to reveal character traits through the choice of subplan a character agent makes [Bah15]. Results have shown that users will ascribe different character traits to different subplans. For instance, users may view Smith as particularly persistent, due to his ability to develop and execute new plans in the face of complications. This is a promising area of research, though outside the scope of this work’s goals.
My third type of story branch is where agents adopt new goals that are subgoals of the original goal. In an intentional plan, this can occur when goal conditions of the planning problem are achieved before the final step of a subplan. This allows a suplan to be reduced in length, though the original goal would not be reached. In yet another modified version of the Prison problem, Smith’s goal is to be atHome(Smith) and needs a bus ticket along with the warden’s approval to get exonerated then travel home. However, the warden denies his request and Smith responds by asking the superintendent. While this will allow him to Testify, be exonerated, and reach the problem goal, he will still need to find a way home.

**Definition 47 (Subgoal Story Branch)** A subgoal story branch, SUBGOAL\((\pi, \chi, \pi')\), is when the reconsidered intention, \(I^R\), in \(\pi\) is not an element of the new plan’s intentions, \(I^R \notin I(\pi')\), and an effect of a step in \(I^R\)’s subplan is in the agent goals of \(\pi'\).

From a semantics point of view, an agent might do this to be opportunistic. By executing as many steps of the subplan as possible, they reduce the effort to achieve their goal and put themselves in a position to take advantage of emergent situations that help them achieve their goals. In terms of intentional representation, unachieved goals due to causal-link threats are modeled as conflict between agents. The CPOCL representation was developed for just this purpose and is well validated among human subjects.

Lastly, in quadrant 4 I have intention revision story branches. This type of story branch contains an intention revision between two plans, meaning both an agent’s goal and subplan of the intention are different. In the Prison example, I refer back to the original plan where Smith adopts the escaped(Smith)
in the new plan when his trial is denied, see Figure 5.9. I direct interested readers to the full details of the representation and evaluation in Section 3.2.

**Definition 48 (Intention Revision Story Branch)** An intention revision story branch, $\text{IR}(\pi, \chi, \pi')$, is where there exists an intention revision (see Def. 25) between $\pi$ and $\pi'$.

The key difference of an intention revision story branch relative to the other three branch types is the cognitive effort required to integrate it into a mental model. An intention revision’s change in goal and subplan initiates a new local goal hierarchy in a user’s QKS, whereas the other story branch types do not. Effectively this change in local goal hierarchy operationalizes the concept of a *filter of admissability* from original theories of intention [Bra03].

### 5.2 Generating Story Branches

My goal is to generate new intentional plans in response to unexpected user actions (exceptional actions) that initiate a *reconsidered intention*. The algorithm I describe in this section uses the solution space from the current state to obtain the available classes. I use the Intention Dependency Graph (IDG) to obtain a characterization of the solution space and use this characterization as input to the deliberation steps from my definition of an *intention revision* (Def. 25). I use this algorithm to generate plans with intention revision for use in human subject evaluations. I simulate exceptional actions during plan execution and restrict new intentional plans to ones that use intention revision story branches, $\text{IR}(\pi, \chi, \pi')$. Once the
final step of a plan is executed, all the intentional plans I have generated are aggregated into a set of story branches. With this set, I generate simple text of non-interactive stories that will contain intention revision as a branching mechanism. These texts are used to evaluate how intention revision affects subject experience in Section 5.3.

Recall the original Prison plan and planning problem in Figures 5.5 and 5.6, respectively, which I use as inputs $\pi$ and $\Phi$ to Alg. 5. In an interactive narrative, Smith would be a (non-player) character agent and the warden would be the user’s agent. After some initialization in lines 1 to 5, I enter the main loop and execute the character agent actions $s_1$–$s_3$ of $\pi$ in lines 6 to 10.

When I reach the first user action, $s_4$ in $\pi$, I enter the Else block at line 11. Instead of executing $s_4$ (ApproveTrial) in line 12, I identify an applicable exceptional action $s'_4$ (DenyTrial) that introduces a causal-link threat $\neg\text{hasTrial}(\text{Smith})$ to $s_5$ (Testify). This causal-link threat causes the non-player agent Smith to reconsider his intention of $\text{exonerate}(\text{Smith})$.

After adding $\text{EFF}(s'_4)$ to the execution state and marking $s'_4$ as EXECUTED (lines 13–14), I create a new planning problem ($\Phi'$ on line 14) with the $\text{currentState}$ as the initial state and other elements equal to the original planning problem, $\Phi$. Using the intention dependency graph (IDG) from Chapter 4, I generate a solution plan-set $\Pi'$ to $\Phi'$ that represents the solution space of configurations of causally-necessary intentions (line 16).

With the solution plans in $\Pi'$, I can determine what the most appropriate story branch is, according to
Figure 5.6 The Prison planning problem $\Phi$ and its elements. This is the same figure as Figure 3.1, re-introduced here for readability.

BDI logic. Lines 17 and 21 test whether the reconsidered intention’s goal is still achievable by way of a duplicate achievement or subplan story branch, $\text{DUP}(\pi, \chi, \pi')$ and $\text{SUBPLAN}(\pi, \chi, \pi')$ respectively. If so, I can use a story branch where no agent goals change and remain consistent with Cohen and Levesque’s concept of a commitment to an intention. Once I determine that a goal change must take place, that is no solution plan in $\Pi'$ exists that contains the same agent-goals as $\pi$, I search $\Pi'$ for a plan that contains a subgoal of the reconsidered intention (line 19). Finally, should no subgoal exist, in line 23 I search for a solution plan $\pi'$ to $\Pi'$, such that it satisfies my definition of an intention revision with $\pi$.

For the example in Figure 5.9, I find a plan with a single action ($\text{Escape}$) that comprises a single intention frame that I use for the intention revision story branch.

In line 26 I account for a solution space that may not contain a plan that adheres to my story branch definitions. While this is undesirable, it still leaves an experience manager with two options. The first is
to end the story after the exceptional action. The experience manager could execute as many actions as possible, and then emphasize the connection with the action. A second approach would be to use a plan from \( \Pi' \) that is satisficing. This could take the form of choosing a plan with minimal agent-goal changes between the original plan and story branch and relying on the concept of anticipated retrospection. This concept can intuitively be described as ‘It will all make sense in the end’ and is thought of as process readers do to tolerate periods of incoherence in a plot [Bro92]. In fact, several acclaimed major motion pictures (e.g. *Hoosiers* [MGM86]) have incoherence (e.g. plot holes) that are never resolved. In the algorithm I describe, I opt for the former, simply executing plan steps until it is no longer possible. I leave the latter strategy of using satisficing plans and their effects on user experience to future work that I discuss further in Chapter 6.

Once I have determined the story branch type, I mark the current plan as having branched and I continue execution with \( \pi' \) as the current plan using a recursive call to \texttt{executePlan} (lines 27–28). The recursive call enables further branching using the same exceptional user action-intention revision mechanism and conveniently aggregates all story branches together (line 30) after the final step is executed.

With the output from Algorithm 5 I can glean some insights into the user’s experience of a branching story. Since \( \Psi \) contains all the story branches a user experienced, I can use it to describe the entire branching structure a user experienced and make comparisons between different experiences.

**Definition 49 (Story Branch Summary)** A Story Branch Summary of a set of story branches, \( \text{Summary}(\Psi) \), is an \( n \)-length character string of D,P,G,I representing the duplicate, subplan, subgoal, and intention revision story branches in a \( \Psi \) with \( n \) branches. Each character also has a subscript representing the step number of the exceptional action \( \chi \) that initiated the story branch addition.

The simple *Prison* example does not capture the full representational power of this definition, as it is simply \( I_4 \). However, I will provide slightly more complex examples in my experiment where the advantages are more clear.

### 5.3 Evaluation

Story branches are an important feature of interactive narrative systems as they enable an experience manager (EM) to create a story branch in response to user actions that would otherwise end the story. My taxonomy of story branches and algorithm for generating them in Sections 5.1 and 5.2 give us the precision required to assess the long-held assumption that branching stories improve a user’s experience. In this section, I describe an experiment who’s goal is to establish the efficacy of using intention revision as a branching mechanism for interactive narrative. My experiment investigates how story branches created with intention revision affect a reader’s experience in narrative texts. Because I use text and not an interactive narrative with choices, I isolate the effects of intention revision so as to not conflate them
Algorithm 5 executePlan(\(\pi, \Phi\)) captures the relationship between a plan, an exceptional action, and generation of plans that introduce intention revision.

**Input:** A plan \(\pi\) with elements \(S(\text{steps}), B(\text{bindings}), O(\text{orderings}), L(\text{causal-links}), I(\text{Intention Frames})\) is a solution to the planning problem \(\Phi\) with elements \(I(\text{Initial state}), G(\text{goal conditions}), A(\text{Agents}), O(\text{Objects}), \Lambda(\text{Actions})\)

**Output:** \(\Psi\), a set of story branches used to generate a branching narrative

1: currentState = \(I(\Phi)\)
2: hasBranched = False
3: \(\Psi = \{\}\)
4: \(\psi = \text{null}\)
5: \(i = 0\)
6: while \(\neg\)hasBranched do
7:  \(s = s_i \in S(\pi)\)
8:  if \(\text{AGENT}(s) \neq \text{user}\) then
9:     currentState += \(\text{EFF}(s)\)
10:    \(s.\text{isExecuted} = \text{True}\)
11:  else \(\triangleright\) Introduce an exceptional action that initiates a reconsidered intention (Def. 21)
12:     \(\chi = s' \in \Lambda(\Phi) : \text{PRE}(s') \in \text{currentState}, e \in \text{EFF}(s')\) introduces a causal-link threat
13:     currentState += \(\text{EFF}(\chi)\)
14:     \(\chi.\text{isExecuted} = \text{True}\)
15:     \(\Phi' = \langle \text{currentState}, \text{G}(\Phi), \text{A}(\Phi), \text{O}(\Phi), \Lambda(\Phi) \rangle\)
16:     \(\Pi' = \text{Sol}(\text{IDG}(\Phi'))\)
17:     if \(\exists \pi' \in \Pi' : \text{DUP}(\pi, \chi, \pi')\) then
18:         \(\psi = \langle \pi, \chi, \pi', \text{DUP} \rangle\)
19:     else if \(\exists \pi' \in \Pi' : \text{SUBPLAN}(\pi, \chi, \pi')\) then
20:         \(\psi = \langle \pi, \chi, \pi', \text{SUBPLAN} \rangle\)
21:     else if \(\exists \pi' \in \Pi' : \text{SUBGOAL}(\pi, \chi, \pi')\) then
22:         \(\psi = \langle \pi, \chi, \pi', \text{SUBGOAL} \rangle\)
23:     else if \(\exists \pi' \in \Pi' : \text{IR}(\pi, \chi, \pi')\) then
24:         \(\psi = \langle \pi, \chi, \pi', \text{IR} \rangle\)
25:     else \(\triangleright\) No story branch classes exist in \(\Pi'\), keep executing \(\pi\) until no longer possible
26:         hasBranched = \(\text{True}\)
27:     \(\Psi = \text{executePlan}(\pi', \Phi')\)
28:     \(i++\)
29: return \(\{\psi\} \cup \Psi\)
with effects from an interactive environment. To measure user experience I use measures of narrative transportation and perceived interest. The results of my experiment enable future work where I can determine the effects of using intention revision as a branching mechanism in an interactive narrative.

5.3.1 Method

5.3.1.1 Story Structures

I designed a longer version of the Prison example as a problem in PDDL, \( \Phi_{\text{Jail}} \), and generated an initial solution plan, \( \pi_{\text{C}}^{3} \)Jail, as input to Algorithm 5. This initial solution is for my control group where non-player character agents achieve three intentions without issue and does not contain intention revisions. This type of plan represents the story from an interactive narrative where the user would have executed three actions consistent with \( \pi_{\text{C}}^{3} \)Jail. My algorithm in Algorithm 5 introduces three user actions that require Smith to revise his intentions and generates plans in response that when aggregated form \( \pi_{\text{T}}^{3} \)Jail. This type of plan represents the story from an interactive narrative where a user chooses actions that require an experience manager to generate a new story branch using intention revision.

Both \( \pi_{\text{C}}^{3} \)Jail and \( \pi_{\text{T}}^{3} \)Jail were designed to allow variants that contain a different number of achieved and revised intentions. The variants are \( \pi_{\text{C}}^{1} \)Jail, \( \pi_{\text{C}}^{2} \)Jail and \( \pi_{\text{T}}^{1} \)Jail, \( \pi_{\text{T}}^{2} \)Jail and are represented in Figure 5.7.

Finally, I designed a second planning problem to account for an alternative intention revision structure (though one that still conforms to my definitions). In \( \pi_{\text{Jail}} \) treatment groups, non-player character agents revise their intentions and subsequently achieve the new revised intention. Conversely, in my second intention revision structure, “Wilderness” (\( \pi_{\text{Wild}} \)), the character agent must revise their new intention three times before they finally achieve an intention. The difference in structure of both stories is captured in Fig. 5.7.

My experiments focus on the causal and intentional structure of intention revision, so I desire to minimize discourse effects (such as re-ordering steps for dramatic effect). After linearizing the partial orderings of a plan’s steps, I apply a simple template, \( \langle \text{agent} 1, \text{action}, \text{agent} 2 \rangle \), that generates a single story sentence for each plan step. Templates are perhaps the most basic form of natural language generation, as they omit intermediate representations (e.g. sentence plans) used by more robust systems [RD97]. Instead, template methods map non-linguistic input (e.g. plan steps) directly to text structure (e.g. story statements). An example of the output from this approach used in the experiment is presented in Figure 5.8. Full text statements for each story are presented in Tables 5.3–5.6.

My experiment was designed to answer the following four research questions:

R1. How does plan-based intention revision affect narrative transportation and perceived interest?

R2. How does plan length affect narrative transportation and perceived interest?

R3. How does plan length interact with the number of intention revisions to affect narrative transportation and perceived interest?
Figure 5.7 Intention revision story structure
The underlined text denotes the exceptional action that initiates an intention revision.

R4. How does the story structure affect narrative transportation and perceived interest?

The answers to these questions will provide insights into how appropriate it is to use intention revision as a branching mechanism for accommodating exceptional user actions.

5.3.1.2 Subjects

Data were collected from 664 participants who were recruited on the CrowdFlower crowd sourcing platform and paid $0.40 USD to read a narrative text and complete the narrative transportation scale (TS, Table 5.1) and the perceived interest questionnaire (PIQ, Table 5.2). Results of a power analysis indicated that the sample size was adequate for finding the desired effects at 0.05 significance level.

5.3.1.3 Procedures

Upon logging into the online experiment, participants were randomly assigned to either the control or treatment groups, and then assigned to read either the Jail or Wilderness story. The control group read a variation of the story where the character agents achieve every intention they commit to. Conversely, the treatment group read stories where the character agent’s original intention was made impossible and had to revise their intentions. In addition, participants were randomly assigned to read a story length of either 15, 25, or 35 statements. I used these story lengths based on the creative constraints required to author a story long enough to have a measurable effect size (should it exist) and the scale of the Glaive planner to generate a long story.

Participants began the experiment by completing a brief demographic survey and then read their assigned story, which was presented as a single page of text. After reading their respective stories, participants completed the fourteen-item transportation scale (TS, Table 5.1) and the twelve-item perceived interest questionnaire (PIQ, Table 5.2).

I hypothesize that when subjects read a story statement that introduces the possibility of an intention revision through an exceptional action, it will compel them to think divergently about a story character’s future intentions. This activity requires recalling story facts and reasoning over them, and creating knowledge structures such as the intention revision QKS subgraph (Section 3.5), all of which increase
Table 5.1 Transportation Scale (TS) items and subscales, (R) indicates reverse coding.

<table>
<thead>
<tr>
<th>Number</th>
<th>Item</th>
<th>Subscale</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>While I was reading the narrative, I could easily picture the events in it taking place.</td>
<td>Cognitive</td>
</tr>
<tr>
<td>2</td>
<td>While I was reading the narrative, activity going on in the room around me was on my mind. (R)</td>
<td>Cognitive</td>
</tr>
<tr>
<td>3</td>
<td>I could picture myself in the scene of the events described in the narrative.</td>
<td>Cognitive</td>
</tr>
<tr>
<td>4</td>
<td>I was mentally involved in the narrative while reading it.</td>
<td>Cognitive</td>
</tr>
<tr>
<td>5</td>
<td>After finishing the narrative, I found it easy to put it out of my mind. (R)</td>
<td>Affective</td>
</tr>
<tr>
<td>6</td>
<td>I wanted to learn how the narrative ended.</td>
<td>Affective</td>
</tr>
<tr>
<td>7</td>
<td>The narrative affected me emotionally.</td>
<td>Affective</td>
</tr>
<tr>
<td>8</td>
<td>I found myself thinking of ways the narrative could have turned out differently.</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>I found my mind wandering while reading the narrative. (R)</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>The events in the narrative are relevant to my everyday life.</td>
<td>Affective</td>
</tr>
<tr>
<td>11</td>
<td>The events in the narrative have changed my life.</td>
<td>Affective</td>
</tr>
<tr>
<td>12-Jail</td>
<td>While reading the narrative I had a vivid image of Smith</td>
<td>Imagery</td>
</tr>
<tr>
<td>13-Jail</td>
<td>While reading the narrative I had a vivid image of the warden</td>
<td>Imagery</td>
</tr>
<tr>
<td>14-Jail</td>
<td>While reading the narrative I had a vivid image of Smith’s family</td>
<td>Imagery</td>
</tr>
<tr>
<td>12-Wild</td>
<td>While reading the narrative I had a vivid image of Jaber</td>
<td>Imagery</td>
</tr>
<tr>
<td>13-Wild</td>
<td>While reading the narrative I had a vivid image of Ahmad</td>
<td>Imagery</td>
</tr>
<tr>
<td>14-Wild</td>
<td>While reading the narrative I had a vivid image of the Bedouins</td>
<td>Imagery</td>
</tr>
</tbody>
</table>

Table 5.2 Perceived Interest Questionnaire (PIQ) items

<table>
<thead>
<tr>
<th>Number</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I thought the story was very interesting</td>
</tr>
<tr>
<td>2</td>
<td>I’d like to discuss this story with others at some point</td>
</tr>
<tr>
<td>3</td>
<td>I would read this story again if I had the chance</td>
</tr>
<tr>
<td>4</td>
<td>I got caught up in the story without trying to</td>
</tr>
<tr>
<td>5</td>
<td>I’ll probably think about the implications of this story for some time to come</td>
</tr>
<tr>
<td>6</td>
<td>I think most people I know would be interested in this story</td>
</tr>
<tr>
<td>7</td>
<td>I liked this story a lot</td>
</tr>
<tr>
<td>8</td>
<td>I would like to read this story again</td>
</tr>
<tr>
<td>9</td>
<td>The story was one of the most interesting things I’ve read in a long time</td>
</tr>
<tr>
<td>10</td>
<td>I would like to know more about why the author wrote the story</td>
</tr>
</tbody>
</table>
Table 5.3 Jail text generated from plans with no intention revision.

<table>
<thead>
<tr>
<th>#</th>
<th>Group ($\pi_{\text{Jail}}$)</th>
<th>Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith is a wrongfully convicted accountant who wants to be exonerated.</td>
</tr>
<tr>
<td>2</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith finds an escape plan.</td>
</tr>
<tr>
<td>3</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith discards the escape plan.</td>
</tr>
<tr>
<td>4</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith complies with demands from the warden.</td>
</tr>
<tr>
<td>5</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith files taxes for the warden.</td>
</tr>
<tr>
<td>6</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith is forced to embezzle money for the warden.</td>
</tr>
<tr>
<td>7</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith makes friends with some inmates.</td>
</tr>
<tr>
<td>8</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith learns his friend has some exoneration evidence.</td>
</tr>
<tr>
<td>9</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith request exoneration from the warden.</td>
</tr>
<tr>
<td>10</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>The warden says he will help Smith get exonerated.</td>
</tr>
<tr>
<td>11</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>The warden gets a new trial for Smith.</td>
</tr>
<tr>
<td>12</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith is assigned a lawyer.</td>
</tr>
<tr>
<td>13</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith convinces his inmate friend to testify.</td>
</tr>
<tr>
<td>14</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith is exonerated by the judge.</td>
</tr>
<tr>
<td>15</td>
<td>$\pi_{C1}, \pi_{C2}, \pi_{C3}$</td>
<td>Smith is released from prison.</td>
</tr>
<tr>
<td>16</td>
<td>$\pi_{C2}, \pi_{C3}$</td>
<td>Smith plans journey home.</td>
</tr>
<tr>
<td>17</td>
<td>$\pi_{C2}, \pi_{C3}$</td>
<td>Smith arranges transportation.</td>
</tr>
<tr>
<td>18</td>
<td>$\pi_{C2}, \pi_{C3}$</td>
<td>Smith gets supplies.</td>
</tr>
<tr>
<td>19</td>
<td>$\pi_{C2}, \pi_{C3}$</td>
<td>Smith travels home.</td>
</tr>
<tr>
<td>20</td>
<td>$\pi_{C2}, \pi_{C3}$</td>
<td>Smith spends time with his family.</td>
</tr>
<tr>
<td>21</td>
<td>$\pi_{C2}, \pi_{C3}$</td>
<td>Smith has a welcome home party.</td>
</tr>
<tr>
<td>22</td>
<td>$\pi_{C2}, \pi_{C3}$</td>
<td>Smith sees his old friends.</td>
</tr>
<tr>
<td>23</td>
<td>$\pi_{C2}, \pi_{C3}$</td>
<td>Smith reminisces about the good old days.</td>
</tr>
<tr>
<td>24</td>
<td>$\pi_{C2}, \pi_{C3}$</td>
<td>Smith enjoys his new freedom.</td>
</tr>
<tr>
<td>25</td>
<td>$\pi_{C3}$</td>
<td>Smith plans to seek justice on the warden.</td>
</tr>
<tr>
<td>26</td>
<td>$\pi_{C3}$</td>
<td>Smith hires a lawyer.</td>
</tr>
<tr>
<td>27</td>
<td>$\pi_{C3}$</td>
<td>Smith files a lawsuit against the warden.</td>
</tr>
<tr>
<td>28</td>
<td>$\pi_{C3}$</td>
<td>Smith gets a trial date.</td>
</tr>
<tr>
<td>29</td>
<td>$\pi_{C3}$</td>
<td>Smith testifies he was forced to embezzle.</td>
</tr>
<tr>
<td>30</td>
<td>$\pi_{C3}$</td>
<td>Smith is cross examined.</td>
</tr>
<tr>
<td>31</td>
<td>$\pi_{C3}$</td>
<td>The warden testifies Smith was the real mastermind.</td>
</tr>
<tr>
<td>32</td>
<td>$\pi_{C3}$</td>
<td>The warden jury deliberates their verdict.</td>
</tr>
<tr>
<td>33</td>
<td>$\pi_{C3}$</td>
<td>The warden is convicted of embezzling.</td>
</tr>
<tr>
<td>34</td>
<td>$\pi_{C3}$</td>
<td>Smith enjoys the justice he brought on the warden.</td>
</tr>
</tbody>
</table>
Table 5.4 Jail text generated from plans with intention revision.

<table>
<thead>
<tr>
<th>#</th>
<th>Group (π_{jail})</th>
<th>Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith is a wrongfully convicted accountant who wants to be exonerated.</td>
</tr>
<tr>
<td>2</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith finds escape plan.</td>
</tr>
<tr>
<td>3</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith discards escape plan.</td>
</tr>
<tr>
<td>4</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith complies with demands from the warden.</td>
</tr>
<tr>
<td>5</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith files taxes for the warden.</td>
</tr>
<tr>
<td>6</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith is forced to embezzles money for the warden.</td>
</tr>
<tr>
<td>7</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith makes friends with some inmates.</td>
</tr>
<tr>
<td>8</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith learns his friend has some exoneration evidence.</td>
</tr>
<tr>
<td>9</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith requests exoneration from the warden.</td>
</tr>
<tr>
<td>10</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>The warden says he will never allow Smith to be exonerated.</td>
</tr>
<tr>
<td>11</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith retrieves the escape plan.</td>
</tr>
<tr>
<td>12</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith takes a spoon from the cafeteria.</td>
</tr>
<tr>
<td>13</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith hangs artwork in his cell.</td>
</tr>
<tr>
<td>14</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith digs a tunnel with the spoon behind the artwork.</td>
</tr>
<tr>
<td>15</td>
<td>π_{jail}^1, π_{jail}^2, π_{jail}^3</td>
<td>Smith escapes through the tunnel.</td>
</tr>
<tr>
<td>16</td>
<td>π_{jail}^2, π_{jail}^3</td>
<td>Smith plans journey home.</td>
</tr>
<tr>
<td>17</td>
<td>π_{jail}^2, π_{jail}^3</td>
<td>Smith arranges transportation.</td>
</tr>
<tr>
<td>18</td>
<td>π_{jail}^2, π_{jail}^3</td>
<td>Smith gets supplies.</td>
</tr>
<tr>
<td>19</td>
<td>π_{jail}^2, π_{jail}^3</td>
<td>Smith travels home.</td>
</tr>
<tr>
<td>20</td>
<td>π_{jail}^{3}, π_{jail}^5</td>
<td>Smith discovers the warden is holding his family for ransom.</td>
</tr>
<tr>
<td>21</td>
<td>π_{jail}^2, π_{jail}^5</td>
<td>Smith follows the directions on the ransom note.</td>
</tr>
<tr>
<td>22</td>
<td>π_{jail}^{3}, π_{jail}^5</td>
<td>Smith turns himself in.</td>
</tr>
<tr>
<td>23</td>
<td>π_{jail}^{3}, π_{jail}^5</td>
<td>Smith is sent back to jail.</td>
</tr>
<tr>
<td>24</td>
<td>π_{jail}^{3}, π_{jail}^5</td>
<td>Smith serves additional prison time for his escape.</td>
</tr>
<tr>
<td>25</td>
<td>π_{jail}^3</td>
<td>Smith plans to seek justice on the warden.</td>
</tr>
<tr>
<td>26</td>
<td>π_{jail}^3</td>
<td>Smith hires a lawyer.</td>
</tr>
<tr>
<td>27</td>
<td>π_{jail}^3</td>
<td>Smith files a lawsuit against the warden.</td>
</tr>
<tr>
<td>28</td>
<td>π_{jail}^3</td>
<td>Smith learns the judge has been bribed by the warden.</td>
</tr>
<tr>
<td>29</td>
<td>π_{jail}^3</td>
<td>Smith plans to serve out prison sentence.</td>
</tr>
<tr>
<td>30</td>
<td>π_{jail}^3</td>
<td>Smith joins a prison gang.</td>
</tr>
<tr>
<td>31</td>
<td>π_{jail}^3</td>
<td>Smith runs progressively more illicit activities.</td>
</tr>
<tr>
<td>32</td>
<td>π_{jail}^3</td>
<td>Smith rises up the prison gang hierarchy.</td>
</tr>
<tr>
<td>33</td>
<td>π_{jail}^3</td>
<td>Smith becomes the leader of the prison gang.</td>
</tr>
<tr>
<td>34</td>
<td>π_{jail}^3</td>
<td>Smith lives out his days in comfortable notoriety.</td>
</tr>
</tbody>
</table>
Table 5.5 Wilderness text generated from plans with no intention revision.

<table>
<thead>
<tr>
<th>#</th>
<th>Group ($\pi_{\text{Wild}}$)</th>
<th>Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber has a job to guide Ahmad from the bazaar to the ivory at an elephant</td>
</tr>
<tr>
<td></td>
<td></td>
<td>graveyard.</td>
</tr>
<tr>
<td>2</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber gets medical supplies.</td>
</tr>
<tr>
<td>3</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber buys food supplies.</td>
</tr>
<tr>
<td>4</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber gathers camping supplies.</td>
</tr>
<tr>
<td>5</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber gets ivory extraction supplies.</td>
</tr>
<tr>
<td>6</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber packs the camels with all the supplies.</td>
</tr>
<tr>
<td>7</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber guides Ahmad through a jungle.</td>
</tr>
<tr>
<td>8</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber guides Ahmad to a desert.</td>
</tr>
<tr>
<td>9</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber and Ahmad set up camp in the desert.</td>
</tr>
<tr>
<td>10</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber and Ahmad spend the night in the desert.</td>
</tr>
<tr>
<td>11</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber and Ahmad gather the supplies.</td>
</tr>
<tr>
<td>12</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber and Ahmad pack the camels with the supplies.</td>
</tr>
<tr>
<td>13</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber and Ahmad leave the desert camp.</td>
</tr>
<tr>
<td>14</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber guides Ahmad through the canyons.</td>
</tr>
<tr>
<td>15</td>
<td>$\pi^1, \pi^2, \pi^3$</td>
<td>Jaber and Ahmad arrive at the elephant graveyard.</td>
</tr>
<tr>
<td>16</td>
<td>$\pi^2, \pi^3$</td>
<td>Jaber and Ahmad unpack the ivory extraction supplies.</td>
</tr>
<tr>
<td>17</td>
<td>$\pi^2, \pi^3$</td>
<td>Ahmad Jaber begins extracting the ivory.</td>
</tr>
<tr>
<td>18</td>
<td>$\pi^3$</td>
<td>Jaber fully loads the camels with the ivory.</td>
</tr>
<tr>
<td>19</td>
<td>$\pi^2, \pi^3$</td>
<td>Jaber and Ahmad leave the elephant graveyard.</td>
</tr>
<tr>
<td>20</td>
<td>$\pi^2, \pi^3$</td>
<td>Jaber guides Ahmad back through the canyons.</td>
</tr>
<tr>
<td>21</td>
<td>$\pi^2, \pi^3$</td>
<td>Jaber guides Ahmad back through the desert.</td>
</tr>
<tr>
<td>22</td>
<td>$\pi^2, \pi^3$</td>
<td>Jaber guides Ahmad back through the jungle.</td>
</tr>
<tr>
<td>23</td>
<td>$\pi^2, \pi^3$</td>
<td>Jaber and Ahmad get exhausted from the terrain.</td>
</tr>
<tr>
<td>24</td>
<td>$\pi^2, \pi^3$</td>
<td>Jaber guides Ahmad and the camels to an oasis.</td>
</tr>
<tr>
<td>25</td>
<td>$\pi^3$</td>
<td>Jaber and Ahmad build a campfire.</td>
</tr>
<tr>
<td>26</td>
<td>$\pi^3$</td>
<td>Jaber and Ahmad eat campfire food.</td>
</tr>
<tr>
<td>27</td>
<td>$\pi^3$</td>
<td>Jaber and Ahmad sing to keep spirits high.</td>
</tr>
<tr>
<td>28</td>
<td>$\pi^3$</td>
<td>Jaber and Ahmad leave the oasis.</td>
</tr>
<tr>
<td>29</td>
<td>$\pi^3$</td>
<td>Jaber and Ahmad arrive at the bazaar.</td>
</tr>
<tr>
<td>30</td>
<td>$\pi^3$</td>
<td>Jaber and Ahmad unpack the ivory.</td>
</tr>
<tr>
<td>31</td>
<td>$\pi^3$</td>
<td>Ahmad sets up a bazaar booth.</td>
</tr>
<tr>
<td>32</td>
<td>$\pi^3$</td>
<td>Ahmad spends all day haggling with buyers.</td>
</tr>
<tr>
<td>33</td>
<td>$\pi^3$</td>
<td>Ahmad sells the final pieces of ivory.</td>
</tr>
<tr>
<td>34</td>
<td>$\pi^3$</td>
<td>Ahmad pays Jaber a fair percentage of the ivory sales.</td>
</tr>
</tbody>
</table>
Table 5.6 Wilderness text generated from plans with intention revision.

<table>
<thead>
<tr>
<th>#</th>
<th>Group ($\pi_{\text{Wild}}$)</th>
<th>Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber has a job to guide Ahmad from the bazaar to the ivory at an elephant graveyard.</td>
</tr>
<tr>
<td>2</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber gets ivory extraction supplies.</td>
</tr>
<tr>
<td>3</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber gets medical supplies.</td>
</tr>
<tr>
<td>4</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber buys food supplies.</td>
</tr>
<tr>
<td>5</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber gathers camping supplies.</td>
</tr>
<tr>
<td>6</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber packs the camels with all the supplies.</td>
</tr>
<tr>
<td>7</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber guides Ahmad through a jungle.</td>
</tr>
<tr>
<td>8</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber guides Ahmad to a desert.</td>
</tr>
<tr>
<td>9</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber and Ahmad set up camp in the desert.</td>
</tr>
<tr>
<td>10</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Bedouins steal the camels from Jaber and Ahmad.</td>
</tr>
<tr>
<td>11</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber and Ahmad plan to return to the bazaar on foot.</td>
</tr>
<tr>
<td>12</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber and Ahmad salvage the supplies.</td>
</tr>
<tr>
<td>13</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber and Ahmad pack the salvaged supplies.</td>
</tr>
<tr>
<td>14</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber and Ahmad leave the desert.</td>
</tr>
<tr>
<td>15</td>
<td>$\pi^{T_1}, \pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber guides Ahmad through the jungle.</td>
</tr>
<tr>
<td>16</td>
<td>$\pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber covers their tracks.</td>
</tr>
<tr>
<td>17</td>
<td>$\pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber and Ahmad are exhausted from the terrain.</td>
</tr>
<tr>
<td>18</td>
<td>$\pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber guides Ahmad to an oasis.</td>
</tr>
<tr>
<td>19</td>
<td>$\pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber and Ahmad set up camp at the oasis.</td>
</tr>
<tr>
<td>20</td>
<td>$\pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber slips and breaks his foot.</td>
</tr>
<tr>
<td>21</td>
<td>$\pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber plans to heal at the oasis.</td>
</tr>
<tr>
<td>22</td>
<td>$\pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber splits the remaining supplies with Ahmad.</td>
</tr>
<tr>
<td>23</td>
<td>$\pi^{T_2}, \pi^{T_3}$</td>
<td>Jaber draws a map for Ahmad.</td>
</tr>
<tr>
<td>24</td>
<td>$\pi^{T_2}, \pi^{T_3}$</td>
<td>Ahmad continues to the bazaar to get help for Jaber.</td>
</tr>
<tr>
<td>25</td>
<td>$\pi^{T_3}$</td>
<td>Jaber builds a campfire.</td>
</tr>
<tr>
<td>26</td>
<td>$\pi^{T_3}$</td>
<td>Jaber sings to keep spirits high.</td>
</tr>
<tr>
<td>27</td>
<td>$\pi^{T_3}$</td>
<td>Jaber eats campfire food.</td>
</tr>
<tr>
<td>28</td>
<td>$\pi^{T_3}$</td>
<td>Jaber cares for his foot.</td>
</tr>
<tr>
<td>29</td>
<td>$\pi^{T_3}$</td>
<td>Jaber discovers Ahmad has stolen all the supplies.</td>
</tr>
<tr>
<td>30</td>
<td>$\pi^{T_3}$</td>
<td>Jaber plans revenge on Ahmad.</td>
</tr>
<tr>
<td>31</td>
<td>$\pi^{T_3}$</td>
<td>Jaber summons all his strength.</td>
</tr>
<tr>
<td>32</td>
<td>$\pi^{T_3}$</td>
<td>Jaber tracks Ahmad.</td>
</tr>
<tr>
<td>33</td>
<td>$\pi^{T_3}$</td>
<td>Jaber captures Ahmad.</td>
</tr>
<tr>
<td>34</td>
<td>$\pi^{T_3}$</td>
<td>Jaber trades Ahmad for supplies with the Bedouins.</td>
</tr>
</tbody>
</table>
Table 5.7 Story branch summaries of text read by experimental groups

<table>
<thead>
<tr>
<th>#</th>
<th>Group</th>
<th>Story Branch Summary (Summary(Ψ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>π₁₀ᵣ</td>
<td>Jail n/a</td>
</tr>
<tr>
<td>2</td>
<td>π₂₀ᵣ</td>
<td>Jail n/a</td>
</tr>
<tr>
<td>3</td>
<td>π₃₀ᵣ</td>
<td>Jail n/a</td>
</tr>
<tr>
<td>4</td>
<td>π₁₁ᵣ</td>
<td>Jail I₁₀</td>
</tr>
<tr>
<td>5</td>
<td>π₁₂ᵣ</td>
<td>Jail I₁₀I₂₁</td>
</tr>
<tr>
<td>6</td>
<td>π₁₃ᵣ</td>
<td>Jail I₁₀I₂₁I₂₉</td>
</tr>
<tr>
<td>7</td>
<td>π₁₁ᵣ</td>
<td>Jail n/a</td>
</tr>
<tr>
<td>8</td>
<td>π₂₁ᵣ</td>
<td>Jail n/a</td>
</tr>
<tr>
<td>9</td>
<td>π₃₁ᵣ</td>
<td>Jail n/a</td>
</tr>
<tr>
<td>10</td>
<td>π₁₁ᵣ</td>
<td>Jail I₁₀</td>
</tr>
<tr>
<td>11</td>
<td>π₂₁ᵣ</td>
<td>Jail I₁₀I₁₅</td>
</tr>
<tr>
<td>12</td>
<td>π₃₁ᵣ</td>
<td>Jail I₁₀I₁₅I₂₅</td>
</tr>
</tbody>
</table>

Table 5.8 The table below contains the mean Transportation Scale (TS) and Perceived Interest Questionnaire (PIQ) scores, with variance in parentheses, for the control (C) and treatment (T) experimental groups of each Jail and Wilderness story variant.

<table>
<thead>
<tr>
<th></th>
<th>Jail</th>
<th>Wilderness</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>15 statements</td>
<td>25 statements</td>
</tr>
<tr>
<td>TS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>46.33 (10.54)</td>
<td>48.84 (14.06)</td>
</tr>
<tr>
<td>T</td>
<td>48.62 (11.65)</td>
<td>53.46 (12.50)</td>
</tr>
<tr>
<td>PIQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>27.49 (8.25)</td>
<td>26.56 (10.42)</td>
</tr>
<tr>
<td>T</td>
<td>28.91 (8.77)</td>
<td>31.63 (7.74)</td>
</tr>
</tbody>
</table>

the degree that users become involved and interested in the plot and characters. Both the TS and PIQ are designed to measure such effects and are widely used with high degree of success (see Section 2.7 for a longer discussion).

5.3.2 Results

Prior to computing inferential statistical analyses, data were inspected and coded to identify missing values, to identify outliers and to ensure that the data were normally distributed. No extreme values were identified. Research questions were tested using a 2 (group: treatment vs. control) x 2 (story structure: Jail vs. Wilderness) x 3 (story length: 15, 25, 35 statements) between subject ANOVA. Unless otherwise noted, all analyses were computed using a critical alpha value of \( p = .05 \). For all data, Levene’s tests confirmed homogeneity of variance. Means and standard deviations for the different conditions are presented in Table 5.8.

R1. How does plan-based intention revision affect narrative transportation and perceived in-
terest?

**Narrative Transportation.** My first research question asked whether plan-based intention affected perceptions of narrative transportation. To address this question I compared narrative transportation scores between participants in the treatment and control conditions. Results of a 2x2x3 ANOVA showed a main effect for group $F(1, 652) = 10.74, p < .01$ where individuals in the treatment condition ($M = 50.70; SD = 12.81$) who read narrative texts that included intention revisions reported higher levels of narrative transportation compared to individuals in the control group ($M = 47.43; SD = 12.92$).

**Perceived Interest.** Regarding perceived interest, a significant main effect of group $F(1, 652) = 6.43, p < .05$ was observed that showed individuals in the treatment condition ($M = 28.51; SD = 8.97$) who read narrative texts that included intention revisions reported higher levels of perceived interest compared to individuals in the control group ($M = 26.67; SD = 9.22$) who read narrative texts that did not include intention revisions.

R2. How does plan length affect narrative transportation and perceived interest?

**Narrative Transportation.** My second research question asked whether story length impacted perceptions of narrative transportation. To address this question I compared mean narrative transportation scores between participants who read stories that were either 15, 25, or 35 statements long. Results showed a significant main effect of story length: $F(2, 652) = 10.14, p < .001$. Post hoc analyses showed that perceptions of narrative transportation increased significantly as story length increased, such that participants who read the 35 statement narrative text ($M = 51.64, SD = 13.33$) and 25 statement text ($M = 49.54; SD = 13.29$) reported higher levels of narrative transportation compared to the participants who read the 15 statement text ($M = 46.27, SD = 11.73$).

**Perceived Interest.** Regarding perceived interest, a significant main effect of plan length was observed: $F(2, 652) = 3.34, p < .05$. Post hoc analyses showed that participants who read the 35 statement narrative text ($M = 28.52, SD = 9.14$) reported higher levels of perceived interest in the story compared to the participants who read the 15 statement text ($M = 26.34, SD = 8.64$).

R3. How does plan length interact with the number of intention revisions to affect narrative transportation?

**Narrative Transportation.** My third research question asked if intention revisions significantly affected narrative transportation and if this relation depended on story length. To address this question I examined the interaction between group and story length however the results failed to show a significant interaction: $F(2, 652) = .60, non-significant$. Further inspection of the results shows that means trended in the expected direction such that narrative transportation increased at a greater rate for the treatment group ($M = 48.15, 51.29, 52.21$) compared to the control group ($M = 48.07, 49.64, 50.19$) as story length increased.

**Perceived Interest.** I conducted a similar analysis to determine whether group and story length interacted to impact perceived interest. Results of this analysis revealed there was no significant interaction between group and story length: $F(2, 652) = .48, non-significant$.

R4. How does the story structure affect narrative transportation and perceived interest?
Narrative Transportation. I addressed the last research question with a planned comparison that examined whether the type of story structure (Jail or Wilderness) affected perceptions of narrative transportation. Participants in the control condition were excluded from this analysis. Results showed that participants who read the Jail story with intention revisions reported higher levels of narrative transportation ($M = 52.21; SD = 12.24$) than participants who read the Wilderness story with intention revisions ($M = 49.17; SD = 13.23$), $F(1, 332) = 4.99, p < .05$.

Perceived Interest. Regarding the impact of intention revision structure on perceived interest, a planned comparison that excluded participants in the control condition showed that participants in the treatment conditions who read the Jail text reported higher levels of perceived interest ($M = 30.20; SD = 8.32$) than participants who read the Wilderness text ($M = 26.79; SD = 9.30$): $F(1, 332) = 12.75, p < .001$.

5.4 Discussion

Creating a story branch in response to user actions is an attractive feature of interactive narrative EMs. When using an intentional plan representation for the story structure, it is essential to maintain the coherence of the goal-oriented behavior of BDI agents in a story branch. I have formalized a taxonomy of coherent story branches that supports the deliberation steps of intention revision. This formalizing of story branches as intention revision deliberation steps ensures that non-player agents in an intentional plan make believable behavior changes. Using this formalization, an accommodation algorithm and the intention dependency graph (IDG) to characterize the solution space, I investigated the long-held assumption that branching stories improve user experience. I evaluated the effects of intention revision story branches by measuring the impact my generated branching stories have on widely used and highly cited instruments for narrative experience: the transportation scale (TS) and the perceived interest questionnaire (PIQ).

To answer my four research questions (R1–4, Section 5.3.1.1), I generated a set of branching stories by simulating player actions that disrupt (non-player) character agents’ intentions, who then revise their intentions. After translating this plan-set into text, human subjects were assigned to conditions and read stories that either included or excluded intention revisions, varied in length, and varied in story structure. Subjects then completed the TS and PIQ. I answered R1 by showing that the treatment group reported significantly higher narrative transportation and perceived interest scores than the control group across two story structures. In response to R2, my analysis showed that perceptions of narrative transportation and perceived interest increased linearly as story length increased. My analysis for R3 showed that there was no interaction effect between group and story length for either narrative transportation or perceived interest, though the results suggested narrative transportation grew stronger for longer stories that included intention revisions. A complicating factor in showing the interaction effect could have been that I inadvertently introduced an agent intention that was interpreted by subjects as betrayal and revenge, which I hypothesize led to an increase in transportation in the control groups. Finally, to address R4 I found that the Jail story, in which the story structure allowed the main character to achieve multiple goals,
led to higher perceptions of narrative transportation and perceived interest. Overall, I conclude that using intention revision as a story branching mechanism is an effective strategy, as it not only continues a story when it would otherwise end but actually increases subjective experience.

There are two immediate areas for future work. The first is to evaluate an interactive narrative that allows users to choose actions instead of simulating exceptional actions. This would enable me to measure the interaction effects between intention revision and an interactive environment. The second is a plan-based model of revenge. I observed an increase in transportation in both the control and treatment conditions when revenge was inadvertently introduced. I view revenge as a more specific type of intention revision and formalizing it would allow further experimentation. Lastly, plan-based intention revision enables a longer-term goal of integrating intentional agent behavior into cognitive architectures that can manage goal lifecycles (e.g. [Cox17]).
Figure 5.9 Mean narrative transportation and perceived interest scores
Interactive narrative is a powerful platform that engages our cognitive abilities through computational models of narrative phenomena. Its applications are wide ranging from education, to entertainment, and even to business (e.g. strategic foresight). Using the IPOCL representation to structure story events ensures that non-player character agents will execute goal-oriented and believable behavior consistent with QUEST knowledge structures. IPOCL and its extension, CPOCL, have leveraged this property to create computational models of suspense, conflict, and false beliefs. One of the more ambitious goals of interactive narrative is for an experience manager, an agent who ensures certain experience qualities, to adapt the story structure to accommodate user actions that introduce inconsistencies to a story’s plot. Adapting the story structure in this way, which I refer to as a story branch, creates a virtual environment where a user is free to exact their agency. The underlying assumption is that by creating a story branch to fit the state of the story environment, including the user’s actions, a user will experience greater agency and with it greater engagement.

However, enabling agency by accommodating user actions is at odds with intentional plans from the IPOCL representation. User actions can introduce causal link threats to the planned steps of a non-player character agent, making an agent’s goal impossible to achieve. When individual agents can’t achieve their goals, the plan will not execute to completion, eroding user agency. For an experience manager to mediate the current state of the story world and the planned experience in the intentional plan, it needs to create a story branch. The story branch is a new intentional plan that accounts for a user’s mental model created by the original intentional plan. Specifically, local goal hierarchies in the QKS that are
representative of character agents pursuing their intentions need to be accounted for. If a character revises their intention in the story branch without a reason salient to the user, what is gained in user agency will be lost through inconsistencies in user mental models. Without a strategy to transition between local goal hierarchies, an experience manager risks introducing inconsistent character agent behavior in a user’s QKS, leading to plot holes or other unwanted experiences, and ultimately to a decrease in engagement.

To address this limitation of an experience manager, I have operationalized intention revision for intentional plans. From this formalization, I developed algorithms to characterize the solution space and to create story branch classes consistent with a user’s QUEST knowledge structure. With these algorithms I show that user engagement increases when users read generated narrative texts that contain intention revision, according to measures of narrative transportation and perceived interest. I discuss these contributions in more detail in Section 6.1 and the exciting areas for future work they have led to in Section 6.2. Lastly, I discuss some new ideas of where this work could lead in the future.

6.1 Contributions

We, as intentional agents, revise our intentions frequently, indicating a range of conditions that initiate when this might occur. Since the first inquiries into intention, scholars have acknowledged the need to identify conditions for when to revise intentions, but there is an absence of formal characterizations. This dissertation has focused on a single, auspicious case that creates sufficient conditions for intention revision; when it is assumed it would be beneficial to a user’s experience. That is, character agents revise their intentions when the story could not proceed any further due to causal link threats introduced by the user. In service of the first operationalization of intention revision for intentional plans, this dissertation has made contributions in three areas that I discuss below. Together these contributions provide an opportunity to enhance a user’s experience, with both a richer model of character behavior for story branches and the ability of an experience manager to provide the agency that is such a desirable property of interactive narrative.

6.1.1 Intention Revision

This first formalization of intention revision for intentional plans is inspired by the BDI control loop from Rao and Goergeff [RG98] and the concept of P-GOALS from Cohen and Levesque [CL90]. The first concept I define is that of a reconsidered intention. When to reconsider an intention is a complex issue and has been the subject of previous study. For an interactive narrative context, the causal link(s) introduced by user actions are the initiators of when an agent reconsiders an intention. Once an intention is reconsidered, I define a three-step deliberation that an agent performs. First an agent assesses whether they can perform an alternate action sequence (subplan) to achieve their same goal. Second, if no subplan exists they identify a new goal. Lastly, they determine if this new goal is a subgoal of the original goal.
Together the reconsidered intention and deliberation process operationalize intention revision and provide an experience manager with a principled way to change an agent’s behavior when creating a story branch.

Successfully applying intention revision in interactive narrative largely depends on user experience. I ensure that this formalization models user comprehension by using the QUEST cognitive model. The QUEST knowledge structure (QKS) is limited in that previous evaluations did not evaluate user comprehension of dropping or revising goals. The QKS limitation stems from the primitive node and edge types that do not represent revised or dropped intentions explicitly. To address this limitation I defined a QKS subgraph that captures the causal and intentional relationships that in aggregate are representative of a user’s mental model after experiencing my definition of plan-based intention revision. Using the QUEST question-answer protocol, I identified three questions that would confirm the QKS intention revision subgraph as representative of a user’s mental model.

Experimental results confirmed that my formalized model of intention revision created the QKS intention revision subgraph, which indicates that users of an interactive narrative would comprehend and understand why a character agent would pursue another goal. This result is significant in its own right, as the first instance of using a QKS subgraph to evaluate comprehension of narrative phenomena. Additionally, this approach has the potential for further work on intention dynamics for BDI agents, of which I provide an example in Section 6.2.1.

### 6.1.2 Intention Dependencies

In my definition of intention revision, there are three steps that make up the deliberation process once an intention is being reconsidered. The first is determining whether the original goal is unachievable. This can be accomplished by finding a plan with the reconsidered intention in the new planning problem’s solution space. Should no plan be found and the goal be unachievable, the second step selects a new goal, again finding a plan from the solution space. Lastly, the new plan is compared with the subplan of the original goal. This determines if it is a subgoal. All three of the deliberation steps ask different questions of a planning problem’s solution space, but diverse planning is an inefficient and inconsistent method to characterize a solution space. I address this limitation by leveraging a fundamental property of intentional plans to characterize a planning problem’s solution space as sets of interdependent character goals.

Intentional plans require that every step of the plan be part of an agent’s subplan to achieve their goal. This property ensures that either an agent is solely responsible for the conditions that allow them to achieve their goal, or they must rely on other agent subplans to provide them. This interdependency between character goals shapes the solution space at a fundamental level. I identify these intention dependencies before planning begins while the story planning graph is being constructed and represent their relationship in the Intention Dependency Graph (IDG). Once complete, the intention dependency graph contains sets of causally-necessary, interdependent character goals that characterize the solution space in a more representative and efficient way than diverse planning. This characterization also supports
the deliberation process of an agent who is reconsidering an intention and requires knowledge of the solution space’s properties to do so (e.g., whether the solution space contains specific plans).

I evaluated the IDG’s ability to address the efficiency and characterization limitations of diverse planning using the classical diversity measures of intra-plan set diversity and inter-plan set diversity, respectively. Intra-plan set diversity measures how different the plans in a plan set are from one another and is used to evaluate how efficient a planner is at generating diverse solutions. Results showed that using the IDG to direct an intentional planner generated a more diverse plan set and used far fewer plans to do so across the benchmark domains. There was one interesting exception in the *Space* problem, where diverse planners had included character intentions that were not causally-necessary to solve the planning problem. This resulted in a higher diversity than using the IDG and underscored the claim that the IDG only includes interdependent character goals that are necessary for solving the planning problem. While padding solution plans with additional non-interfering character goals can make a plan set more diverse and could be of practical use for deliberation, I view it as a separate problem.

Inter-plan set diversity measures a planner’s ability to identify solutions from different regions of the search space. When I compared IDG-generated solutions to diverse planners using inter-plan set diversity, I found that across the benchmark domains the IDG had solutions that no other diverse planner plan set had found. This suggests that an IDG-generated plan set is appropriate as an accurate characterization of the solution space.

In summary, the IDG is a more efficient and accurate method to characterize a planning problem’s solution space than diverse planning. They key difference between the two methods is that instead of relying on a user to specify an estimate of how many plans they need to obtain a diverse plan set, the IDG analyzes the structure of the planning problem to devise partitions of the solution space. While the results support this approach, an additional insight is that the benchmark problems are not that diverse. The efficiency and accuracy of the IDG could support quick interaction cycles of a problem author in the development of more diverse benchmark problems.

### 6.1.3 Subject Experience of Intention Revision

While I have used QUEST to evaluate human comprehension of the formalized definition of intention revision and developed the algorithmic means to support it with the intention dependency graph, it is also imperative to determine a subject’s experience of intention revision in a narrative context. Intention revision provides the mechanism to create story branches, the assumption being it enables greater user agency, and I have investigated the impacts on overall experience. After all, if user agency comes at a great cost of overall narrative experience, it is worth considering the trade-off.

I have positioned my definition of intention revision as a mechanism for an experience manager to respond to exceptional user actions. In fact, when an agent is executing the deliberation steps, there are several possibilities of plans that do not involve an intention revision. To characterize these
differences, I defined a taxonomy of four story branch classes for intentional plans. The first class, duplicate achievement, captures when the exceptional action and new plan do not change a character’s intentions or subplans to achieve their goals. Second, subplan story branches are where an agent changes their subplan (action sequence) in the new plan to achieve their same goal as the original plan. A third class is subgoal story branches, where an agent chooses a new goal that was an effect of their original subplan. Lastly, intention revision story branches capture when an agent chooses a new goal and conforms to my definition of an intention revision. Together these classes capture the type of story branches a user can experience when using an intentional plan as the story structure.

This taxonomy of story branches enables their use in an interactive narrative. I defined an algorithm that an experience manager would use to create the appropriate story branches to accommodate exceptional user actions. This algorithm takes an intentional plan and problem as input, and uses the intention dependency graph to characterize the solution space from which a branch is selected. The algorithm’s final output is a set of story branches that were created to reach the goal conditions of the original planning problem. The algorithm’s output can be used in a story branch string that I define to summarize the branching structure of the intentional story structure a user experienced. The summary can then be used for comparisons between experiences of an interactive narrative with an intentional plan structure.

Another feature of the aforementioned algorithm for story branches is that it enables me to separate the effects of interactive narrative and the effects of branch structure on the basic experience. Using an intentional plan as the original story, I generated intention revision story branches when a simulated exceptional user action was introduced. Both the original plan and branching plan were translated to text using simple template-based natural language generation, after which they were read by control and experimental groups, respectively. Using measures of Narrative Transportation (TS) and Perceived Interest (PIQ), my results indicate that subjects in the experimental group experience greater TS and PIQ when compared to those in the control group who read texts that did not contain intention revision. These results are encouraging as they suggest that adapting to user actions will not only enable agency but could also independently lead to greater experience.

6.2 Future Work

In this dissertation, I have contributed to formal models of intention revision, applications of cognitive models, solution space exploration, and user experience of story branches. These areas have also led to several interesting and relevant ideas for future work. I discuss three of the most immediate in the next sections.
6.2.1 Computational Model of Betrayal

In my experiment where I generated narrative texts from intention revision plans, I inadvertently introduced a betrayal of trust between the warden and Smith. Upon reflection, betrayal is a more specific type of intention revision that often occurs in narrative. Betrayal could be formalized in intentional plans and evaluated using a QKS subgraph.

The formalization would extend the concepts of shared plans and intentions between agents [GK96]. The concept of shared plans and intentions has been referenced before and investigated. The central idea is that two agents would form a joint plan to achieve an intention, then one of the agents would revise their intention. This agent would execute steps of a subplan that introduces an unresolvable causal link threat to the shared plan, forcing it to be dropped by the other agent.

I could evaluate this model of betrayal in a similar approach to intention revision. The first step would be to define a QKS subgraph that would extend the intention revision subgraph, such as the one in Figure 6.1. Note that instead of $\chi$ making the original goal unachievable as a consequence, it is $\chi'$, an event in the new goal’s local hierarchy. I could validate this subgraph and its construction using the QUEST question-answer protocol.
6.2.2 Procedural Semantics for Domain Authors

In my evaluation of the intention dependency graph (IDG) on the narrative planning benchmarks, I discovered that there are very few intention revision story branches present in the problems. This limits the strength of empirical claims that can be made about the IDG. For instance, I was able to generate a solution plan for every partition problem from the IDG even though I hypothesize that there are cases when a partition problem has no solution plan. Demonstrating this, along with other properties such as many intention revision story branches in a benchmark domain would help make stronger claims about the IDG’s ability to characterize a solution space.

To address this limitation and develop narrative benchmarks with a diverse solution space, domain authors need to be more tightly connected to the solution space. An approach to this is to use the IDG as a domain author tool. In addition to the solution vertices, the IDG structure contains the full ancestry of vertices combined to achieve them. This structure could be used by procedural semantics (procedures for determining truth values) that enable a domain author to use different logic formalism to reason over the solution space.

In Figure 6.2, the IDG from my example is now represented for use by procedural semantics. Note that the levels have been removed and the direction of the arrows reversed from the IDG in Figure 4.4. Transforming the IDG into this representation would enable modal reasoning to determine agent intentions essential to all solutions. In the example, I could state that the Smith, exonerate(Smith) agent-goal pair is
necessary in every solution with the modal statement $v_{\pi} \models \Box(\text{Smith, exonerate} (\text{Smith}))$ in KT45 modal logic. When this statement is true, it helps a domain author understand the importance of a single character agent, as it is present across all possible solutions. This example demonstrates the usefulness of the IDG in designing more diverse domains for a more comprehensive set of narrative benchmark problems.

### 6.2.3 Subject Experience of Intention Revision

My evaluation of plan-based intention revision in narrative text showed that readers experienced greater narrative transportation and perceived interest when reading narrative texts that contained intention revisions compared to texts without them. This insight lays the foundation to explore the effects of intention revision in an interactive narrative environment.

In a similar experimental design to the one described in Section 5.3, subjects would be recruited on a crowd-sourcing website to read and interact with narrative texts, then complete the narrative transportation and perceived interest instruments. Subjects would be assigned to an interactive group and non-interactive group so that I could perform yoking. This allows comparisons between subjects who receive the same information (narrative text), isolating the interactive effect as the interactive group is able to choose their actions. To enable the choices an interactive group could take, the existing Jail story would have to be augmented to enable a full set of experimental groups, as represented in Figure 6.3. This experiment would provide evidence of the effects intention revision has in an interactive environment.

An additional change to the Jail text would be the removal of the betrayal and revenge elements. While they were introduced inadvertently and are more fine-grained forms of intention revision, I believe they affected the narrative transportation and perceived interest scores more than the coarser-grained intention revisions used elsewhere in the stories. Their removal will ensure a more targeted and accurate measurement of intention revision in interactive narrative.
Figure 6.3 Interactive narrative experimental groups, an extension to the experiment represented in Figure 5.7.
BIBLIOGRAPHY


