A Use of Flashback and Foreshadowing for Surprise Arousal in Narrative Using a Plan-Based Approach

Byung-Chull Bae[1] and R. Michael Young[1]

[1]Liquid Narrative Group, Department of Computer Science, North Carolina State University
Raleigh, NC 27695, USA
bbae@ncsu.edu, young@csc.ncsu.edu

Abstract. This paper describes work currently in progress to develop a computational method for generating flashback and foreshadowing, specifically targeted at the evocation of surprise in the reader's mind. Flashback provides a backstory to explain what caused the surprise outcome. Foreshadowing provides an implicit hint about the surprise. Our study focuses on surprise as a cognitive response. The reader's story construction process is simulated by a plan-based reader model that checks unexpectedness and postdictability of the surprise.

1 Introduction

In a narrative, there exist two different levels of time. One is story time experienced by the characters in the story world. The other is discourse time experienced by readers [5]. The events in the story plane are related temporally and causally based on their natural order of occurrence, but they are often rearranged in the discourse plane intentionally In other words, authors can let the readers know about some facts in advance or hide some information until a certain point for a dramatic effect. The narrative theorist Genette [8] explains this temporal disparity using the terms *analepsis* and *prolepsis*. Analepsis tells (or shows) what has happened in the past with respect to the present. Similarly, prolepsis presents what will happen in the future with respect to "now" in the story. The former is like rewinding the story, and the latter is like fastforwarding the story.

In this paper we use the terms *flashback* as an instance of analepsis and *foreshadowing* (or *flashforward*) as that of prolepsis, following the conventions in cinematic media [2][5][17]. Foreshadowing implicitly alludes to a future event in a manner that makes it difficult for the reader recognize its meaning until the event actually happens, In contrast, flashforward explicitly presents a future event in a manner that makes a reader immediately aware of the impending activity[1]. The

---

[1] Based on Genette's terminology, foreshadowing is an example of *advance mention* and flashforward is an example of *advance notice*. The former is implicit; the latter is explicit.

distinction between foreshadowing and flashforward, however, may not be clear in film narratives in which flashforward can be partially explicit with help of the camera (e.g., a shot in which a character's face is hidden by manipulation of the camera angle).

Emotions such as suspense, curiosity, and surprise help the readers focus attention on a story, contributing to the readers' sense of satisfaction [1][18]. These emotions, according to structural affect theory [3][4], can be aroused by manipulation of temporal elements in narrative structure. Empirical studies have shown that temporal manipulation of discourse structure can produce different cognitive and emotional responses by influencing the reader's inferences and anticipation [9][12]. The work described here is motivated by this approach to focus on the use of two narrative devices – flashback and foreshadowing – for surprise arousal in narrative generation.

Although flashback and foreshadowing can be important tools for the author to communicate with the readers for a dramatic effect, methods for the automatic generation of these devices have been rarely investigated in previous computer-based story generation systems.

2 Related Work

In 1976, James Meehan developed a storytelling system called *Tale-Spin,* one of the first attempts at automatic story generation. Since Tale-Spin, a number of story generators have introduced various ways to build (or to present) a better story in terms of creativeness, story comprehension, interest, etc. This section outlines major previous works closely related to our study: use of flashback and foreshadowing in narrative, surprise as a cognitive response from literature, and the efforts to generate computer-based stories with a dramatic effect.

2.1 Use of Flashback and Foreshadowing in Narrative

Both flashback and foreshadowing are narrative devices that present story events out of temporal order. Flashback describes some past events related to the present; foreshadowing gives allusion (possibly implicit) to some future events. Typically in film media, flashback often functions to provide backstory in support of a main story line[2] [17], being presented either as a continuous sequence or as a series of scenes showing only the crux of the backstory. By contrast, foreshadowing, as "hints of what is to come" [5], gives only implicit or partial information. If foreshadowing is completely implicit, the reader realizes its meaning only later, in retrospect. If it is explicit with partial information, the reader is forced to fill in the information gap in her mental representation of the story. This kind of foreshadowing often serves to focus the reader's attention on a specific event.

---

[2] Flashback can also be used to refer to an entire main story. For example, a narrator can tell a main story as a form of flashback in retrospect, often with the first person prospective. In this study, we focus only on flashback that functions to describe a backstory.

In interactive narrative-centered virtual environments, a user's interaction with a virtual character can prompt flashback or foreshadowing (or flashforward). For example, flashback can depict a character's recall of his or her own past events; flashforward can present a character's imagination of possible outcomes [13]. Our approach is different in that we dynamically generate flashback and foreshadowing from the story at the discourse level.

2.2 Postdictable Surprise as a Reader's Emotion as Literary Response

We read books not just to acquire information but also to receive some kind of reward or to stimulate interest through reading [11][18]. According to Oatley [14], a reader's emotions that arise in response to reading can be classified into two types – external and internal. External emotions are evoked as the reader confronts the pattern (i.e., schema or structure) of the narrative; internal emotions are aroused as the reader enters the story world described in the text. Cognitive responses such as curiosity or surprise epitomize external emotions that occur from narrative structure. Empathy with characters in a story is an example of internal emotions. This classification is in accord with Kintsch's distinction between *cognitive interest* and *emotional interest* [11]. The former arises from a well-organized discourse structure; the latter from emotional context in the story. In our present study we focus only on surprise as cognitive interest rather than as an emotional interest or an internal emotion.

Excluding emotional interest, Kintsch introduces the notion of *postdictability* that can contribute to the value of cognitive interest regardless of story types [11]. Postdictability characterizes a story structure in which every part makes sense for the reader as a whole so that she can construct "a coherent macrostructure" with no conflict in retrospect. Surprise or curiosity based on unexpected events, without consideration of postdictability, will produce no interest to the reader because it is likely to fail to contribute to building a coherent story as a whole in the reader's mind. For this reason, surprise associated with unexpected important story outcome should be postdictable [11] [18].

2.3 Computational Model of Story Generation with Dramatic Effects

Our study is primarily motivated by two previous story generation systems: Minstrel [21] and Suspenser [6]. Minstrel employs various creative writing techniques such as generation of suspense, tragedy, and foreshadowing. Particularly, Minstrel aims to create a sense of inevitability and a sense of unity in the story by foreshadowing contrived or unexpected events. As a criterion for contrivance, Minstrel uses a notion of uniqueness (or uncommonness). In other words, a story event is contrived or uncommon if Minstrel has no memory of it. While this heuristic has benefits in terms of domain independence and learning, the selection of uncommon or unexpected events remains a difficult problem to solve.

Like Minstrel, Suspenser creates a story with a dramatic effect, but it focuses only on one cognitive interest: suspense. Suspenser selects important actions for inclusion in a story using a plan-based reader model that measures the suspense level of the

reader at a certain point while reading a story. While Suspenser concentrates on the selection of story content to increase suspense level, our system stresses the presentation order of story content.

Based on the efforts made by Minstrel and Suspenser, we focus on building foreshadowing and flashback automatically in computer-based narrative generation using a plan-based approach.

3   System Architecture

In this section we describe Prevoyant, a computational model of flashback and foreshadowing currently under development. Prevoyant will produce as output a story containing structure intended to evoke surprise in the reader's mind. Given a source story described using a plan data structure, Prevoyant determines the content and insertion point for flashback and foreshadowing events. Prevoyant makes use of a reader model which reflects a reader's conception of a story world constructed during reading. The temporally reconstructed story plan is visualized in a 3D virtual world. To visualize the stories Prevoyant produces, we employ Zocalo, web-based intelligent planning services for control of 3D virtual worlds [22]. Fig.1 illustrates the overall input and output process of Prevoyant.
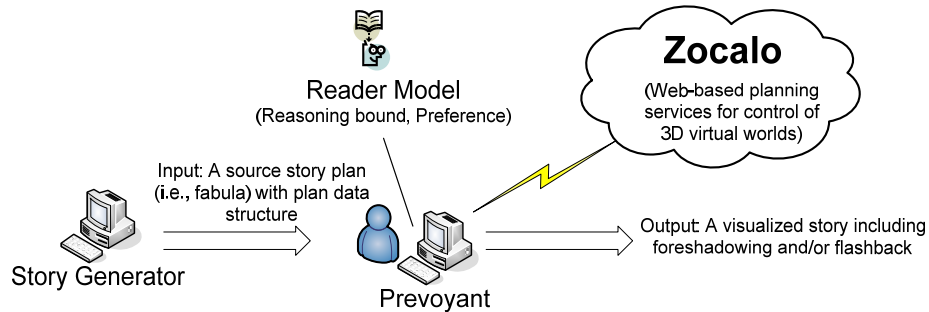
Fig. 1. Input and Output of Prevoyant with the Reader Model and Zocalo Environment

3.1 Generation of Fabula as Source Story

The distinction between *fabula* and *syuzhet*, made by Russian formalism, is a useful concept to understand any narrative element with regard to the ordering of story events [2][8]. The term *fabula* refers to a storyworld in which story events occur in chronological order. The term *syuzhet* characterizes selected contents of the fabula arranged in a certain presentation order, taking into account the reader. To understand a story, the reader constructs her mental representation about a fabula indirectly by drawing inferences about its structure based on an observed syuzhet [2]. In our research, a source story (i.e., a fabula in chronological order) is built by Longbow, a discourse planner employing a partial-order causal-link planning algorithm with

hierarchical action decomposition [23]. This story is then changed to a syuzhet through temporal reconstruction (and possibly through a content selection process).

The plan data structure of the source story created by the Longbow planner includes sets of plan steps, binding constraints, ordering constraints, causal links, and decomposition links. The rich causal relationships in the plan structures created by Longbow correspond closely with the characteristics of the causal networks used by Trabasso and Sperry to represent the relationship between causal relatedness and importance of story events [20]. They claim that the causality among story events is an essential factor for story recall and judgments of important story events [19]. Prevoyant ranks importance of story events based on this causal network story model.

3.2 The Reader Model

Prevoyant's reader model simulates the reasoning process of an *implied reader* that refers to an imaginary, ideal reader as a counterpart to an *implied author* [5][16]. Without the notion of an implied reader, the story may provide readers either with too much explanation or with a sudden change of context [7]. Either can complicate the reader's comprehension process. Prevoyant employs a plan-based reader model using the Longbow planning system, on the basis of a study that has shown that abilities of human planning can be described as those of partial-order planners [15]. Traditionally, AI planners have played a role as a problem solver in task environments. In our approach, we use the Longbow planning system as a reader model in two specific ways: (1) when constructing foreshadowing, (i.e., when a hint about an important story outcome is given), the reader model checks whether or not the reader can infer the outcome as a logical consequence on the basis of her current knowledge; (2) when constructing flashbacks, (i.e., when backstory elements supporting an important story outcome are given), the reader model checks whether or not the reader can comprehend the outcome without any logical conflicts with what she read so far.

The Longbow planning system is based on a domain-independent planning algorithm named DPOCL (Decompositional Partial Order Causal Link) [23], consisting of three components: a *plan-space search* [10] algorithm, a function to represent a user's reasoning resource limits, and a heuristic function used by the planning algorithm to guide the plan-space search process. The process of plan-space search searches through a plan space in which a plan is refined to another plan by fixing a flaw in it. The flaw can be an open precondition (i.e., no actions to achieve this precondition in the plan), a threat (i.e., a step whose effect conflicts with a causal link in the plan), or an abstract step which can be decomposed further. A solution is a complete and consistent plan (i.e., a plan with no open preconditions, no unexpanded abstract steps, no threatened causal links, and no cycles in the temporal ordering constraints). This reader model simulates plot-related inferences performed by the reader. The (possibly partial) plans in the plan space represent completions of the story by the reader. The reasoning bound function defines the limits on the reader's ability to infer plans. The heuristic function characterizes the reader's preference for plans in the planning process such as the preference for content selection. The reader's current knowledge while reading a story is represented by a plan library that

defines a set of operators. This reader's plan library characterizes world knowledge and text knowledge based on what she has read so far.

3.3 Prevoyant

Prevoyant aims to create surprise at an important story outcome, which can make the reader more engaged in the story [1]. For the surprise arousal, Prevoyant uses two narrative techniques: foreshadowing and flashback. Foreshadowing provides the reader with (possibly implicit) anticipation[3]; flashback explains what caused the surprise (i.e., the unexpected outcome) to happen. To meet this end, Prevoyant employs a generate-and-test design incorporating three major components: the generator, the evaluator, and the implementer. The generator and the evaluator work together to reconstruct a given story based on inferences directed by the reader model. Specifically, the evaluator checks unexpectedness and postdictability in terms of surprise arousal. After the reconstruction of story events is complete, the implementer decides manifestation details based on the specific medium in which the story is being told. The subsequent sections explain these three components in more detail.

3.3.1 Generator and Evaluator
Given in chronological order, events in a source story (i.e., a fabula) are temporally reconstructed by the generator and the evaluator in Prevoyant. The generator creates potential flashbacks and foreshadowings. The evaluator checks unexpectedness for the flashbacks and postdictability as a whole, taking into account the reader model.

3.3.1.1 The Generator: Flashback Selection
Flashback provides the reader with a backstory occurring in the past, typically associated with a character, an object, or an event [17]. The use of flashback, however, should be carefully designed because frequent use of flashback, especially to explain some insignificant elements in the main story, may harm the story momentum.

A narrative structure that evokes surprise is characterized by "sudden presentation of an unexpected outcome", where expository or initiating events associated with the outcome are presented after the outcome or even omitted [18]. This narrative structure for surprise arousal is similar to that used for curiosity arousal. Both of these structures present unexpected outcomes without their initiating events, but in the two cases the reader's knowledge about the initiating events is different. If the reader knows that the initiating events are missing or only partially depicted, curiosity occurs; if the reader is not aware of the absence of the initiating events, surprise occurs [1][4][18]. Based on this model, the generator selects flashback events by identifying a Significant Event (SE) and its Initiating Events (IE) in the input story. The algorithm for flashback selection is outlined as follows:

---

[3] This kind of foreshadowing is also known as *Chekhov's Gun*, which is a literary technique known from Chekhov's quote: "one must not put a loaded rifle on the stage if no one is thinking of firing it"

I. Given an input plan representing a story, identify a set of plan steps that directly connect to goal literals in the goal state, assuming that goal literals are important outcomes in the author-centric story generation system[4] [1]. Define this set as a candidate SE set.

II. Consider each SE in the candidate SE set. Let the IE for this set be just a series of events which are causally linked from the initial state to the SE. We say that an IE is *separable* just when the IE can be omitted without causing any open preconditions in the story plan with regard to the events before the SE. Define this pair of the SE and its IE as a separable causal chain. If an SE contains more than one separable causal chain, non-deterministically pick one of these separable causal chains. If there are multiple SEs that have separable causal chains, select the SE with the highest importance rank. The importance is rated in terms of causality, based on the claim that causal connections are more likely to predict recall [20]. The more causal links a candidate SE has, the more significant it is.

III. Define flashback as the IE on the separable causal chain defined in (II), and put its temporal order after the presentation of the SE.

The separability of the IE ensures that the reader does not detect the absence of the IE from the rest of the story until the associated SE happens. Moreover, the generator can select a flashback as a minimum set of the IE with which the reader can comprehend how the surprise (i.e., the unexpected significant event) could happen, minimizing the influence on the story flow. The minimum set of the IE can be determined by appeal to the reader model.

3.3.1.2 The Evaluator: Unexpectedness Check

When the generator identifies a flashback as a set of separable IE, the evaluator confirms the unexpectedness by reasoning about the impact of the omission of the separable IE on the reader's comprehension process. This process is simulated using plan-space search in the reader model. The search process searches for a complete plan to satisfy all the preconditions of the SE, without flashback (i.e., the initiating events on a separable causal chain). If a complete plan is found, the evaluator determines expectedness, that is, the surprise fails. In this case, the generate-and-test process iterates and the generator makes another attempt with a different SE in the candidate SE set. Our present study does not measure the level of unexpectedness. More refined algorithm to achieve qualitative unexpectedness will be necessary. If no complete plan can be found, the evaluator determines unexpectedness.

3.3.1.3 The Generator: Foreshadowing Selection

In the work we describe here, foreshadowing refers to the omitted IE that will be presented after the presentation of the SE as a form of flashback. The use of foreshadowing can enforce postdictability of the omitted events. As mentioned earlier, postdictability is a crucial factor in surprise arousal [11] [18].

Foreshadowing gives hints to the extent that the reader cannot predict its allusion, but can postdict it in retrospect. The implicitness level of foreshadowing will depend

---

[4] In this paper we assume that all goal literals have equal importance.

on story types or the reader's preferences. In our present study, the criterion for foreshadowing is defined as the least allusion to the omitted IE. As a way of the least allusion, Prevoyant employs two methods of partial flashforward. One is to present a character or an object in an IE in case it is not yet introduced before the SE happens. The other is to present an IE with the removal of important parameters by marking them as hidden. Either should be checked through the evaluator and the reader model whether or not it retains the unexpectedness. While this foreshadowing can be put at multiple temporal positions before the presentation of the SE, it is conventionally placed in the beginning of the story, minimizing the interruption of the story flow.

3.3.1.4 The Evaluator: Postdictability Check

After the selection of foreshadowing and flashback is complete, the evaluator checks postdictability. Surprise is postdictable if there is no conflict among the story elements when the reader reconstructs the whole story in retrospect. Since the flashback and foreshadowing in our approach are based on a story plan that is created from a sound planner, the temporally reconstructed story in retrospect guarantees its postdictability.

3.3.2 The Implementer

Given a temporally reconstructed story plan with flashback and foreshadowing, the implementer manifests it in a specific medium. Through the interaction with Zocalo [22], the implementer converts the story plan into a script that consists of a series of executable actions in a commercial game engine, such as the Unreal Engine by Epic Games.

As a way to handle flashback and foreshadowing at the discourse level (or at the text/media level), the implementer takes advantage of patterns of camera usage typically seen in film. When flashback is connected to a character, associated visual cues are often followed by flashback: the camera tends to show the character in close up as a signal to the viewer, and then follow with flashback scenes. Using a different color filter for shots, e.g., sepia, can also be an indication of flashback.

Unlike flashback, foreshadowing is given without a visual cue since it is implicit in nature. In film narrative, the camera serves as an implicit narrator by virtue of controlling what is seen in a scene (or a shot). Specifically, *hidden* information in the foreshadowing can be efficiently handled by the camera. For example, consider a foreshadowing with a primitive action *Gunshot* (?*shooter*, ?*victim*), where ?*shooter* and ?*victim* are binding variables. If the variable ?*shooter* is marked as hidden, the face or identity of the shooter can be concealed through the manipulation of camera angle. If the both variables are supposed to be hidden, the sound of gunfire and the close-up of the gun can be shown without revealing the shooter or the victim.

3.4 The Zocalo Environment

Zocalo is a service-oriented architecture developed by Liquid Narrative group at North Carolina State University, integrating a variety of planning-related software components such as Fletcher (Web Services), Longbow (a Hierarchical Partial-Order

Causal-Link Planner), and the Execution Manager (Intelligent controller of commercial game engine environments) [22]. Based on previous efforts developing the Mimesis system [24], Zocalo provides interface services to translate plan steps in the story plan into executable actions in the 3D virtual environments. By incorporating these service-oriented capabilities of Zocalo, Prevoyant creates stories in 3D graphics. Fig.2 illustrates the Zocalo architecture.
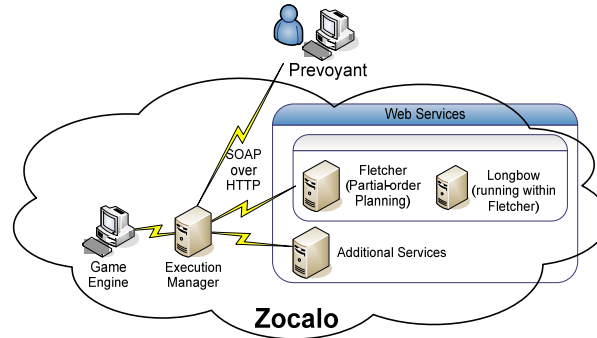

**Fig. 2.** Architecture of Zocalo, a Web-Based Planning Services

4 Example

In this section, we illustrate how an input fabula can be reconstructed to a *syuzhet* with flashback and foreshadowing. As an input fabula, we picked an objective story that was previously used for a pilot study measuring suspense in narrative generation [6]. This *fabula* was created by Crossbow, a C# implementation of Longbow, and shares the same plan structure characteristics. The design of the *fabula* – including a plan library, initial state and goal state specifications, etc. – was designed by a domain engineer who was not involved in our current effort. This input *fabula* contains a background story represented by ground literals in the initial state and a main story consisting of sentences based on 20 plan steps seen in Fig. 3. The plan steps that comprise a complete story plan were created by the Crossbow planner to satisfy goal literals in the goal state.

For the sake of clarity, we represent causal relationships among plan steps, literals in the initial state, and literals in the goal state as illustrated in Fig.4. Plan steps in the *fabula* are represented by circles. The numbers inside the circles denote the plan step numbers in Fig.3, based on causal and temporal constraints in the *fabula*. In this example, step 1 is the opening step; step 20 is the closing step. The directed arcs between two steps denote causal links from their head-step to tail-step. Specifically, the dotted-line arcs denote the causal links starting from the initial state. The literals in the initial/goal state are represented by rectangles. This *fabula* employs eight ground literals in the initial state, and satisfies seven ground literals in the goal state. The thick solid arrow represents that a goal literal is satisfied by an effect of a plan step (e.g., an effect of the step 4 satisfies the goal literal g1).

Background: The lunatic supervillain known as Jack has been developing biological weapons of devastating proportions. To accomplish the final stages of weapon development, he kidnapped the famous scientist, Dr. Cohen, and brought him to his private fortress on Skeleton Island. Jack expected that the FBI would soon send Smith, their top agent, to rescue Dr. Cohen. To keep the troublesome Smith out of his hair, Jack ordered his own agent, Erica, to monitor Smith and capture him if he is assigned to Dr. Cohen's rescue operation.

Story: (1) Erica installs a wiretap in Smith's home while he is away. (2) Erica eavesdrops on the phone conversation in which Smith is given the order to rescue Dr. Cohen. (3) Erica meets with Smith. (4) Erica tells Smith that her father was kidnapped by Jack and taken to Skeleton Island, and she asks Smith to save her father. (5) Erica gives Smith the blueprints of Jack's fortress, with her father's cell marked. (6) Erica provides Smith with a boat for transportation to Skeleton Island. (7) Before going to the island, Smith hides a diamond in his shoe. (8) Smith goes to the port containing Erica's boat. (9) Smith rides the boat to Skeleton Island. (10) Smith sneaks into the cell marked on the map containing Erica's father. (11) Jack and his guard capture Smith as he enters the cell. (12) The guard disarms Smith. (13) The guard locks Smith into the cell. (14) Smith bribes the guard with the diamond in his shoe. (15) The guard unlocks the door. (16) Smith leaves the cell. (17) Smith sneaks to the lab where Dr. Cohen is captured. (18) Smith fights the guards in the lab. (19) Smith takes Dr. Cohen from the lab. (20) Smith and Dr. Cohen ride the boat to shore.

Fig. 3. Input Fabula Story (Adopted from Cheong [6])
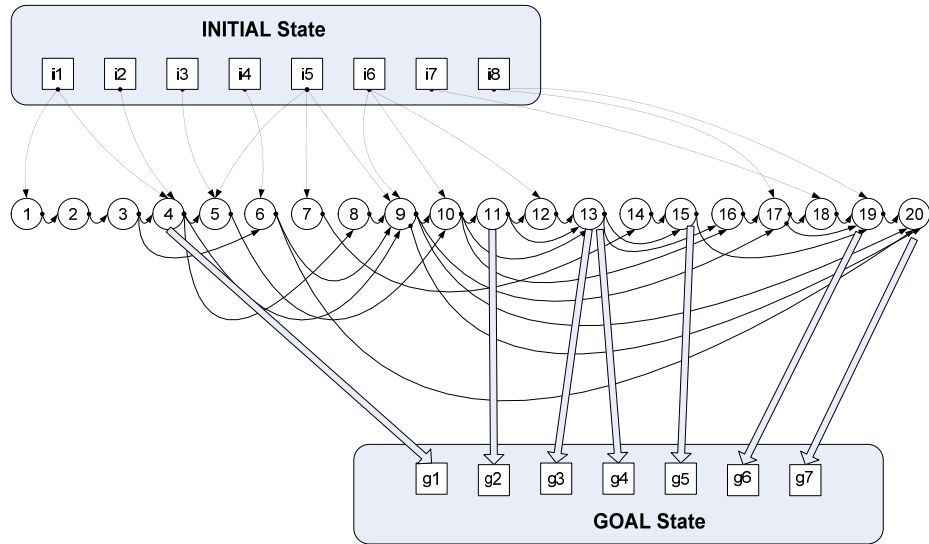
Fig. 4. A Representation of Causal Relationships among Plan Steps, Initial State Literals, and Goal State Literals

According to the algorithm described in the previous section, Prevoyant first selects a candidate Significant Event (SE) set consisting of steps 4, 11, 13, 15, 19, and

20 that directly connect to the goal state. For each candidate SE, Prevoyant checks the possibility of a separable causal chain that will be selected as flashback. For example, step 7 and step 14 can be a separable Initiating Events (IE) as flashback for step 15. The omission of those steps does not cause any open preconditions with regard to the events before step 15. In other words, the reader is not aware of the absence of those events until step 15 is executed. The evaluator confirms its unexpectedness by checking whether or not the reader can infer step 15 without seeing steps 7 and 14. Next, foreshadowing is selected to show partial information of step 7.

In this example, flashback scene consists of two steps: (7) Before going to the island, Smith hides a diamond in his shoe; (14) Smith bribes the guard with the diamond in his shoe. Prevoyant selects step 7, which is farther from step 15 than step 14, for the least allusion. In the story plan, step 7 is instantiated by an action *Hide-Diamond-In-The-Shoe* (*?person*), where the variable *?person* is bound to a constant *Smith*. For implementation of foreshadowing, the camera can show a shot in which someone hides a diamond in his shoe. If step 7 was instantiated with more parameters (e.g., *Hide* (*?Smith*, *?Diamond*, *?Shoes*)), depending on the plan library design, the camera could show either a diamond or shoes in the beginning.

5 Discussion and Future Work

This paper describes work currently in progress to develop a computational method for generating flashback and foreshadowing, specifically targeted at the evocation of surprise in the reader's mind. Our study focuses on surprise as a cognitive response, which is mainly aroused by manipulation of the temporal structure of narrative, rather than as an emotional response. While our present study concentrates on the difference of ordering (of story events) between story time and discourse time, difference of duration or frequency (of story events) between them will be of the same importance [8]. Foreshadowing or flashback, for example, can be presented partially or separately [5]. In that case, the use of foreshadowing and flashback will require careful consideration in order to increase the values of other cognitive interest such as curiosity or suspense at the same time. We plan to take into account the effects of combined cognitive interest (e.g., generation of suspense with surprise outcome or curiosity followed by surprise).

In interactive narrative environments, foreshadowing and flashback will be more closely associated with the cognitive emotion of a story character which a user plays. The content and timing for flashback will be dynamically chosen to reveal the cause of the character's surprise. Moreover, the flashback can show only partial information of the backstory in order to evoke curiosity in the user's mind, which can motivate the user to uncover the whole concealed backstory. Since foreshadowing should be presented prior to the coupled flashback, the system's mediation may occur. In other words, once a foreshadowing is given by predicting a character's surprise, the system will guide the character (that is played by a user) to execute some actions which will lead to the surprise situation. Then the flashback will be given to support the backstory. This kind of flashback can be used to bridge the gap between a main story and its backstories.

References

1. Alwitt, L.: Maintaining Attention to a Narrative Event. In: Shohov, S.P. (eds.) Advances in Psychology Research, vol. 18, pp. 99--114 (2002)
2. Bordwell, D.: Narration in the Fiction Film. University of Wisconsin Press, Madison (1985)
3. Brewer, W.F. and Lichtenstein, E. H.: Event Schemas, Story Schemas, and Story Grammars. In: Long, J. and Baddeley, A. (eds.) Attention and Performance, vol. 9, pp. 363--279 (1981)
4. Brewer, W.F. and Lichtenstein, E. H.: Stories Are to Entertain: A Structural-Affect Theory of Stories. In: Journal of Pragmatics, 6, pp. 473--483 (1982)
5. Chatman, S.: Story and Discourse: Narrative Structure in Fiction and Film. Cornell University Press, Ithaca, NY (1978)
6. Cheong, Y.: A Computational Model of Narrative Generation for Suspense. Ph.D. Dissertation, Department of Computer Science, North Carolina State University (2007)
7. Emmott, C.: Narrative Comprehension: A Discourse Perspective. Oxford University Press, New York (1997)
8. Genette, G.: Narrative Discourse. Cornell University Press, Ithaca, NY (1980)
9. Hoeken, H., van Vliet, M.: Suspense, Curiosity, and Surprise: How Discourse Structure Influences the Affective and Cognitive Processing of a Story. Poetics, 26: 277--286 (2000)
10. Kambhampati, S., Knoblock, C. A., and Qiang, Y.: Planning as refinement search: a unified framework for evaluating design tradeoffs in partial-order planning. Artificial Intelligence, vol. 76, pp. 167--238 (1995)
11. Kintsch, W.: Learning from Text, Levels of Comprehension, or: Why Anyone Would Read a Story Anyway. In: Poetics, vol.9 pp. 87--98 (1980)
12. Levoarto, M. C. and Nemesio, A.: Readers' Responses While Reading a Narrative Text. In: Empirical Studies of the Arts, vol.23 (1), pp. 19--31 (2005)
13. Marsella, S.C., Johnson, W.L., & LaBore, C.: Interactive Pedagogical Drama. In: 4th International Conference on Autonomous Agents, pp. 301--308 (2000)
14. Oatley, K.: A Taxonomy of the Emotions of Literary Response and a Theory of Idenification in Fictional Narrative. Poetics, vol.23, pp.53--74 (1994)
15. Ratterman, M.J., Spector, L., Grafman, J., Levin, H, and Harward, H.: Partial and Total-Order Planning: Evidence from Normal and Prefrontally Damaged Populations. In: Cognitive Science, vol.25 (6), pp.941--975 (2001)
16. Rimmon-Kenan, S.: Narrative Fiction, Routledge, London and New York (2002)
17. Sijll, J. V.: Cinematic Storytelling: The 100 Most Powerful Film Conventions Every Filmmaker Must Know. Michael Wiese Productions, Studio City, CA (2005)
18. Tan, Ed S.: Emotion and the Structure of Narrative Film: Film as an Emotion Machine. L. Erlbaum Associates, Mahwah, NJ (1996)
19. Trabasso, T. and van den Broek, P.W.: Causal Thinking and the Representation of Narrative Events. In: Journal of Memory and Language, vol.24, pp.612--630 (1985)
20. Trabasso, T. and Sperry, L.: Causal Relatedness and Importance of Story Events. In: Journal of Memory and Language, vol.24, pp.595--611 (1985)
21. Turner, S. The Creative Process: A Computer Model of Storytelling and Creativity. L. Erlbaum Associates, Hillsdale, NJ (1994)
22. Vernieri, T. M.: A Web Services Approach to Generating and Using Plans in Configurable Execution Environments, Master Thesis, Department of Computer Science, North Carolina State University (2006)
23. Young, R. M., Pollack, M. E., and Moore, J. D.: Decomposition and Causality in Partial Order Planning. In: Proceedings of the 2nd International Conference on AI and Planning Systems, pp.188-193 (1994)
24. Young, R. M.: An Overview of the Mimesis Architecture: Integrating Intelligent Narrative Control into an Existing Gaming Environment. In: The Working Notes of the AAAI Spring Symposium on Artificial Intelligence and Interactive Environment. (2001)