Intelligent Narrative Technologies: Papers from the 2013 AIIDE Workshop (WS-13-21)

Modelling Character Knowledge in Plan-Based Interactive Narrative to Extend Accomodative Mediation

Justus Robertson and R. Michael Young
Liquid Narrative Group
Department of Computer Science
North Carolina State University
Raleigh, NC 27695

Abstract

In plan-based interactive narrative systems, user actions must be integrated with an ideal story structure in order to strike a reasonable balance between user agency and story quality. A process called *mediation* addresses this issue by selecting appropriate responses to possible detrimental user actions in a given interactive narrative plan. One way the algorithm mediates exceptional user activity is *accommodation*, the process of replanning future events to incorporate an unwanted action and accomplish the story's authorial goals. However, this approach is limited by an underlying assumption that the story world exists as an object outside the user's subjective experience. In this paper, we incorporate a model of user knowledge into plan-based mediation in order to widen the space of plans the algorithm may search during accommodation by allowing the process to replan past events that occur outside a user's imperfect knowledge of the story world.

Introduction

One way a participatory story may be coordinated in an interactive narrative system is to generate a plan structure that all characters are expected to follow. If given sufficient autonomy, a human character may interfere with causal links in this narrative plan and disrupt the desired sequence of events. In order to preserve the user's sense of agency while also immersing her in a meaningful, well-structured story, the system must incorporate some policy for mitigating exceptional user activity. This policy is known as *narrative mediation* (Riedl, Saretto, and Young 2003; Harris 2005; Riedl et al. 2008). Existing mediation techniques align exceptional user actions with desired plot structure by either *intervention*, which disables a user's exceptional action, or *accommodation*, which incorporates the unwanted action into the existing story by replanning future events.

A mediation component that wishes to grant the user a sense of agency prefers to accommodate as many of the user's exceptional actions as possible. Accommodation is preferable to intervention because it allows a user to effect change in a story world while preserving the author's original story objectives. Intervention is less desirable because it changes the story world such that the intended outcome of a user's behavior never occurs. If intervention repeatedly dismisses actions, a user may realize their behavior is limited. Even so, mediation algorithms are forced to intervene when an accommodative solution cannot be found.

One problem that limits the mediation process from using accommodation are situations where no accommodative solution exists. This occurs when a mediative planner exhausts the accommodative search space without finding a valid plan. However, a mediation component with a model of player knowledge can search more accommodative plans by replanning story events that occur both before and after the exceptional action. This can be done as long as the planner maintains story-world consistency by restricting modifications of past events to aspects of the world unobserved by the user. In this paper, we overview a system that widens the accommodative mediation search space by introducing a modified Partial Order Causal Link (POCL) planner that leverages a model of character knowledge to navigate the user's set of possible worlds at plan-time.

Related Work

This work is based on the original *reactive mediation* algorithm (Riedl, Saretto, and Young 2003) created for the Mimesis system (Young et al. 2004). The Mimesis system is an architecture that both generates a narrative plan and executes this plan in a virtual environment. Integrated within Mimesis is the mediation algorithm, which reconciles real-time user actions with a pre-planned plot. When an exceptional step occurs, the mediation system uses a precomputed policy to either *intervene* by disabling the user's exceptional action or *accommodate* the user's behavior. Accommodation allows the system to substitute a new plan that integrates the exceptional action into a new plot that accomplishes the author's goals.

This work focuses on integrating a model of character knowledge into the mediation process. The inclusion of player knowledge in an interactive narrative system was pioneered by the Interactive Drama Architecture (IDA) (Magerko 2006). IDA acts as a director that guides players through an interactive drama. This director utilizes a model of hypothesized user knowledge that it infers from a user's actions in the game world. The director represents the user's knowledge with the Soar architecture's working memory elements (WMEs). This WME database is updated as the player progresses through the story and is used at any point as a hypothesis for what the character knows about the world. This database allows the system to do novel tasks such as accept authorial goals that specify knowledge states instead of external world states.

Unlike IDA, this work uses a model of character knowledge to dynamically change aspects of a story world. This use is similar to Sunshine-Hill and Badler's work on generating alibis for NPCs in open-world simulations (Sunshine-Hill and Badler 2010). Sunshine-Hill and Badler reduce the computational cost of simulating large crowds by introducing "perceptual simulation", a concept that relies on a user's incomplete knowledge of the virtual world to approximate a full simulation. The system selectively simulates observed NPCs in a way that statistically guarantees the approximation is perceptually equivalent to a full simulation. In a similar way, we change aspects of a story world in a way that logically guarantees the world is perceptually equivalent to the original story from the user's perspective.

Algorithm

This paper presents an algorithm that is able to identify what a character will have possibly experienced in a given POCL plan that represents a narrative, remove steps from the plan that are guaranteed to be unobserved by the character, and find an alternate plan (if one exists) that accomplishes the author's goals. It is capable of ordering replanned events before and after the user's exceptional step while ensuring the story-world is consistent from the character's perspective.

Overview

The algorithm takes as input an initial POCL story plan, which is a partial ordering of story events the system has prepared to play out in a virtual environment. It then locates every causal link within the story where the user's character has the opportunity to break the desired flow of events. For each of these causal links the system attempts to create a contingency plan. In order to replan the system first removes every event from the story that is ordered after the user's harmful action, builds a database of the user's potential knowledge, and uses the database to further reduce story events to those the user could possibly observe.

The system passes this information to a specialized partial order planner which only adds events to the story that do not violate the causal flow of anything already presented to the user's character. This planner has the ability to order new story events before the user's exceptional action while guaranteeing the story remains in the user's set of possible worlds. If succesful, the planner returns a valid plan. This plan is stored as a node in a graph that branches for each opportunity the user has to break the story plan. This graph can be traversed by a story manager at execution time as a player makes decisions in the virtual environment.

POCL Planning

This system operates on two data structures: a POCL plan and a set of databases that represent the knowledge of desired characters in a story. POCL plans are comprised of partially ordered steps, which represent actions to be taken by characters in a story world. Possible actions are defined by a set of general action templates, called $operators$. When an operator is added to the plan and modified to talk about particular objects within the domain, it is called a $step$. POCL plans are found using $refinement\ search$. That is, the planner navigates a space of partial plans by identifying and correcting flaws. Once the set of flaws is empty, the planner has found a solution to the planning problem. In addition to a narrative plan, the system must also maintain a database of user knowledge.

Knowledge Annotation

The knowledge annotation algorithm builds a database that represents what a character could possibly know about a story world. Knowledge is characterized on both a per-step and per-condition basis. The knowledge annotation process begins with a complete POCL plan P and for every step s within the plan determines whether the character observes s in any total ordering of P. If s is never observed, the algorithm determines if any effect e from its set of effects are possibly observed.

Our current model of observation is simple: we say a character observes a step if they are possibly at the location where the step happens when it happens. This definition of observation is guaranteed to contain everything a character knows about actions that occur in the story world, but may also contain actions the character has not come to know due to the partial order nature of P. This property of observation guarantees that the world remains consistent for a player, but may reduce the total accommodation space the planner is able to navigate.

In addition to tracking observed story steps, the system must also track observed effects of unobserved actions in order to ensure all aspects of a story world persist properly. The current algorithm tracks only effects that are consistently observed across all total orderings of P. If some effect e is observed as true in some total orderings and false in others, the system must switch e's establishing step from unobserved to observed. Once it is determined that a character has observed a step or effect, the knowledge representation of the step or effect can be added to the character's knowledge database. The completed database will hold all steps and effects a character could possibly observe in the story world.

Policy Generation

In reactive mediation, the $Policy\ Generation$ algorithm is executed before run time. Its purpose is to take the original narrative plan and precompute a policy for handling all possible exceptional user actions that could occur during interaction. This algorithm builds a policy using only accommodation, as opposed to both accommodation and intervention. However, intervention could easily be included when the algorithm is integrated into a full mediation system.

The system has been modified to properly prepare plans for $Bidirectional\ Accommodation$, the method of replanning events before and after a user's exceptional step. The algorithm begins by analyzing each causal link l in the narrative plan. If there exists any possibly performed user step s that could negate the literal that l introduces, the algorithm must prepare a new contingency to handle the exceptional action. It prepares a contingency by generating a new plan using bidirectional accommodation.

The original narrative plan must be modified before bidirectional accommodation can be performed. First, all events ordered explicitly between the exceptional action and goal step are removed from the plan. Next, a database of the user's knowledge of narrative events and conditions is created and used to strip away all unobserved elements preceding the exceptional action. Finally, the algorithm finds the set of flaws introduced into the plan and sends the prepared plan fragment to bidirectional accommodation.

Bidirectional Accommodation

Bidirectional Accommodation is the system's replanning component, able to find new plans that accommodate an exceptional user action identified during policy generation. The algorithm is built on Penberthy and Weld's UCPOP system (Penberthy and Weld 1992). UCPOP has six phases: termination, goal selection, operator selection, subgoal generation, causal link protection, and recursive invocation. Of these six phases, bidirectional accommodation modifies the operator selection phase to ensure new steps and their effects are $consistent$ with the user's knowledge of the story world.

Any new action added to the plan that could be ordered before the user's exceptional step is consistent if it is not possibly observed by the user's character. An effect is consistent if it does not change the character's observed world state in any total ordering of the plan. If the system detects an inconsistent effect e of a consistent action s, it can make the effect consistent by creating new preconditions of $\neg e$ at every step where the effect is observed. However, if e is consistent in some total orderings, but inconsistent in others, s must be considered inconsistent. This situation is similar to the one encountered when annotating observations of effects.

If successful, bidirectional accommodation returns a story plan that accomplishes the author's objectives, incorporates the user's exceptional action, may include modifications to story events that occurred prior to the inciting incident, and remains consistent with everything the user has observed in the story world.

Example

To illustrate the mechanics of bidirectional accommodation and motivate its use in an interactive environment, consider the following scene from the third act of $The\ Dark\ Knight$ (Nolan, 2008). The film's antagonist, the Joker, has been apprehended by the vigilante Batman. During interrogation, the Joker reveals he has kidnapped DA Harvey Dent and Rachel Dawes, who is Batman's love interest. The Joker tells Batman that Rachel is held at Ave. X, Harvey at 252nd St., and that Batman has time to save only one of them. Batman selfishly chooses to pursue Rachel at the expense of Harvey, who is the public figurehead of a lawful Gotham city.

In the film, the Joker lies to Batman who finds Harvey at Ave. X instead of Rachel. It is important that Rachel dies for the final act of the story to commence. If this scene were to play out in an interactive context controlled by reactive mediation where the user plays Batman, the system would not find an accommodative plan for the player's potential choice of moving to 252nd St. where they would find and save Rachel. Instead, it would disable the user's action and force them to travel to Ave. X.

However, with a model of user knowledge the system recognizes the events of the story that place Harvey at Ave. X and Rachel at 252nd St. are outside the user's incomplete knowledge of the story world. These actions are removed from the plan during policy generation and the bidirectional accommodation algorithm is free to find a story that fits Batman's exceptional action of traveling to 252nd St. while ensuring Rachel is the one who perishes in the bomb blast. This restores a choice and sense of local agency to the player and also allows the system to maintain its global goals.

Limitations and Future Work

One limitation of this system is the simplistic model of character knowledge that unnecessarily limits the aspects of the story world that accommodative mediation has the opportunity to rearrange. This model may be improved by adding more complexity to what aspects of the world are presented to the character. We could also examine if everything presented to the character is observed and retained by the player.

This work may also be expanded away from reactive mediation and merged with the proactive mediation algorithm (Harris 2005) that predicts and acts to prevent exceptional user activity before it occurs. Allowing proactive mediation to reason about character knowledge may have similar benefits as reactive mediation.

Otherwise, the largest limitation of the current system arises from the partial ordering of POCL plans. While causal links are useful for determining detrimental user actions, they introduce the complexity of many total orderings instead of a single story line. This complexity limits unobserved events or objects to those determined to be unobserved across all possible total orderings. This problem could be resolved in a forward-chaining planner that tracks causal links, like the Intentional Fast Forward Planner (Ware 2012).

Conclusion

In this paper we presented a simple model of user knowledge and a modification to plan-based mediation that allows the system to search a wider range of potential accommodative plans by restructuring the past as well as the future of a narrative. This is useful because accommodation allows interactive narrative systems to preserve a well-structured story line while also allowing a user to effect change in the story world.

References

Harris, J. 2005. Proactive mediation in plan-based narrative environments. Master's thesis, North Carolina State University.

Magerko, B. 2006. *Player Modeling in the Interactive Drama Architecture*. Ph.D. Dissertation, Dept. of Computer Science, University of Michigan.

Penberthy, S. J., and Weld, D. S. 1992. Ucpop: A sound, complete, partial order planner for ADL. In Nebel, B.; Rich, C.; and Swartout, W., eds., *KR'92. Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference*. San Mateo, California: Morgan Kaufmann. 103–114.

Riedl, M.; Stern, A.; Dini, D.; and Alderman, J. 2008. Dynamic experience management in virtual worlds for entertainment, education, and training. *International Transactions on Systems Science and Applications* 4:23–42.

Riedl, M.; Saretto, C. J.; and Young, R. M. 2003. Managing interaction between users and agents in a multi-agent storytelling environment. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, AAMAS '03, 741–748. New York, NY, USA: ACM.

Sunshine-Hill, B., and Badler, N. 2010. Perceptually realistic behavior through alibi generation. In *Proceedings of the 6th Artificial Intelligence and Interactive Digital Entertainment Conference (AIIDE)*.

Ware, S. G. 2012. The intentional fast-forward narrative planner. In *Eighth Artificial Intelligence and Interactive Digital Entertainment Conference*.

Young, R. M.; Riedl, M. O.; Branly, M.; Jhala, A.; Martin, R. J.; and Saretto, C. J. 2004. An architecture for integrating plan-based behavior generation with interactive game environments. *Journal of Game Development* 1:51–70.