Learning Player Preferences to Inform Delayed Authoring

David Thue¹ and Vadim Bulitko¹ and Marcia Spetch² and Eric Wasylishen¹
¹Department of Computing Science, ²Department of Psychology
University of Alberta
Edmonton, Alberta, Canada, T6G 2E8
{dthue | bulitko | mspetch | ericw}@ualberta.ca

Abstract
Of all forms of Intelligent Narrative, interactive narratives are uniquely well-poised to benefit from player modelling techniques. Given the availability of immediate player feedback as interactions with the narrative’s world, the traditional task of delayed authoring can be informed with the author’s knowledge of which types of players should prefer each event; this allows generated narratives to dynamically adapt and fulfill the player preferences collected by the model. In this paper, we present PaSSAGE (Player-Specific Stories via Automatically Generated Events), our implementation of preference-informed delayed authoring in the setting of interactive entertainment. Recent results from a human user study with 102 participants indicate that for players with minimal previous gaming experience who found the game easy to follow, using our preference-informed techniques can improve their enjoyment of a computer role-playing game.

Introduction
For any commercial venture in Intelligent Narrative to succeed, the resulting product must appeal to a wide audience. Traditional methods for achieving broad appeal typically involve gathering as much audience feedback as possible before release, then striving to reach some compromise between the author’s goals and a wide range of audience preferences. Unfortunately, such compromises often result in content which, while appealing to some members of the audience, may greatly displease others (e.g. Half-Life 2’s driving sequences, or Façade’s social tension) (Valve Corporation 2004; Mateas & Stern 2005). Although this approach is necessary for most forms of narrative, Interactive Narrative allows a mechanism for incorporating audience (player) feedback after release: by automatically observing the player’s interactions with the narrative’s world, her preferences for narrative content can be learned and used by an intelligent narrative system. We refer to this process and its supporting authorial tasks as preference-informed delayed authoring; the author describes which types of players would most enjoy each element of narrative content, and his decisions to present or omit certain elements are made at run-time by the narrative system, based on the player’s feedback thus far. In this paper, we present PaSSAGE (Player-Specific Stories via Automatically Generated Events), our implementation of preference-informed delayed authoring in the setting of interactive entertainment (Thue et al. 2007).

Audience-informed Delayed Authoring
As a decision-making process, the authoring of interactive narratives has many opportunities for automation. With intelligent narrative systems, authors encode their idea of what might happen, along with a general knowledge of how things should happen. Given this information, the narrative system becomes informed; that is, it becomes capable of automatically carrying out authorial decisions at run-time. Although many types of knowledge may be useful in producing a successful narrative (e.g. dramatic theory, narratology), the opportunity for interactive narratives to benefit from player-specific adaptation leads us to focus the following discussion on techniques for gathering and using player feedback as a narrative occurs. We refer to this practice in general as audience-informed delayed authoring.

In Façade, player feedback is gathered from both natural language utterances and in-game gestures (e.g. handling objects, hugging characters, etc.), and recorded as adjustments to a zero-sum affinity value between the game’s two non-player characters (NPCs) (Mateas 2002). By presenting the player with events that encourage her to take sides in arguments between the two NPCs, Façade models how well each NPC likes the player. Based on which NPC holds positive affinity for the player, narrative events become either available or unavailable for execution, ultimately helping to determine the conclusion of the story. While this approach to delayed authoring is certainly audience-informed, it does not consider which of several available narrative events the player might prefer to see happen; ties among available events are broken randomly instead.

In the Interactive Drama Architecture, a player model is specified in advance as a probability distribution over player movements in the narrative’s world (Magerko 2006). Based
on this model and a representation of the narrative’s state, the likelihood of the player violating the preconditions of pending events is estimated and used to initiate subtle (yet still effective), preemptive corrections when future violations seem likely (e.g., having an NPC burst into laughter in a room nearby to draw the player in). Although Magerko’s system is based on a player model, its predictions seem only casually dependent on actions taken by the current player.

In Mirage, player actions are parsed by a rule-based system to build a profile of the personality being portrayed by the player through her in-game avatar (Seif El-Nasr 2007). Mirage’s rules are used to associate a set of author-anticipated player behaviours with adjustments to the personality profile, which is maintained as a vector across five character traits: <reluctant hero, violent, self-interested, coward, truth-seeker>. Unlike Façade, Mirage uses its learned player data to affect the selection priority of narrative events: given an author’s specification of how each event’s priority should be altered based on the portrayed personality profile, Mirage automatically determines the priority of all available events, and chooses the event with the highest priority. In addition to modelling the player character’s personality, Mirage tracks the player’s cursor movements in an attempt to gauge both her inclination to choose an action and the degree of her hesitation in doing so. Inclinations toward actions are used to prompt behaviour changes in non-player characters (such as blocking the player from leaving a room), and hesitation measures are used to regulate the magnitude of adjustments made to the personality profile. Although Mirage informs several parts of its delayed authoring process with audience information, like Façade, it remains targeted toward players who appreciate a well-crafted drama. In the following sections, we argue that in interactive entertainment, dramatic tension is only one of the many sources of enjoyment that players seek.

Following David Kiersey’s theory of temperaments, Gómez-Gauchía and Peinado attempt to automatically customize the behaviour of their narratives’ NPCs using a Case-Based Reasoning approach (Gómez-Gauchía & Peinado 2006). Before the game begins, players fill out a questionnaire: applying Kiersey’s theory to the result then indicates their temperament as a proportional combination of four basic types (e.g. <artisan: 20%, guardian: 50%, idealist: 10%, rational: 20%>). Given a knowledge base of dialogue and behaviour variations designed for several player temperaments, Gómez-Gauchía and Peinado’s system selects the variation whose associated temperament most closely resembles the current player’s. Variations include altering the politeness of the NPC’s comments and the speed of the NPC’s movements. If the proportions of the selected variation’s four basic types fail to match the player’s temperament exactly, the system adapts the variation by further adjusting both dialogue politeness and NPC movement speed within restricted ranges. Although the modelling process of Gómez-Gauchía and Peinado’s system is independent of the player’s actions in-game, one can envision an extension in which a Kiersey-inspired questionnaire could be encoded within the narrative’s events, allowing the more seamless technique of learning the player’s temperament on-line.

In recent work, Sharma et al. extend Nelson and Mateas’s work in Declarative Optimization-based Drama Management to incorporate the modelling of player preferences (Sharma et al. 2007; Nelson et al. 2006). Through a post-game player survey, ratings were obtained for their overall interest in the presented narrative, their interest in each of the narrative’s individual events, and their degree of confidence in their other ratings. By combining the feedback from each play with a trace of its narrative events, Sharma et al.’s player model creates a set of cases for a Case-Based Reasoning system, to be used by their drama manager in associating the current player’s trace of events with interest ratings for potential future events. Subsequent events are selected via an expectimax search to maximize both a set of predefined author interests and the model’s estimated player interests. While Sharma et al.’s work is strongly directed toward informing delayed authoring with player preferences, their dependence on a preexisting case base of player interests limits their flexibility post-release. Instead of relying on gathering enough pre-release feedback to sufficiently inform the model (or perhaps redistributing post-release feedback via the Internet), we advocate a method of automatically acquiring and using the player’s feedback while they play, independent of other players’ data.

Similar to Sharma et al.’s technique of optimizing for player interests, Barber and Kudenko’s recent work presents a narrative planner designed to bring about social dilemmas that are expected to be of high interest to the current player (e.g. betraying a friend to achieve great personal gain) (Barber & Kudenko 2007). Set in the context of an interactive soap opera, their system models the player as numerical values across several personality traits: <honesty, faithfulness, selfishness, etc.>. The model is updated via author-assigned value adjustments that are attached to each dilemma choice (e.g. +1 honesty, -1 selfishness), and it is used to predict which choice the player will most likely select when presented with a particular dilemma. Given a set of author-defined interest values for each dilemma’s choices, the system uses the model’s prediction to select the dilemma that maximizes the likelihood of having an interesting outcome for the current player. Barber and Kudenko’s work is very near to fitting our definition of preference-informed delayed authoring; while it succeeds in both learning and using player feedback on-line, its use of single, general values for the expected interest of dilemma choices neglects the fact that players with different personalities will likely have different levels of interest in each dilemma choice. Estimating each choice’s interest values for various personality models is reflected in part of our approach.

Preference-informed Delayed Authoring

We now present PaSSAGE, our implementation of preference-informed delayed authoring in the context of a computer role-playing game. Following previous work by Peinado and Gervas, PaSSAGE models player preferences as a vector of values for five types of players that Laws identifies as being useful for storytellers in pen and paper role-playing games (e.g. <Fighter (F) = 41, Method Actor (M) = 41, Storyteller (S) = 101, Tactician (T) =
Encounter Creation

In PaSSAGE, all narrative events are encounters - events which directly involve the player. As such, each encounter always offers one or more opportunities for the player to act in response; we refer to each such opportunity as a branch of the encounter. During the creation of an encounter, in addition to specifying the action that takes place, authors annotate each branch with information concerning what types of players would most enjoy playing along that branch. For example, given an encounter wherein a murderer’s identity is revealed, one possible branch might be to inform the local authorities of his location in exchange for a reward, while a second branch might involve a direct vigilante attack. For the former branch, the author may be writing for Tacticians or Power Gamers, while the latter branch might be preferred by Fighters. Branch annotations are made by encoding values in a vector similar to the one maintained by et al. (2007).

Learning Player Preferences

To learn its model of the player’s preferred styles of play, PaSSAGE must be informed with a set of rules indicating which player types (if any) are indicated by a given player action. These rules can be encounter-dependent (e.g., “if the player demands a reward for locating the murderer, boost the player model’s inclination toward the Power Gamer type (+100)”), or encounter-independent (e.g., “if the player attacks a friendly NPC, advance the player model along the Fighter type (+40)”). An example of such a model would be: M = 0, S = 0, T = 4, P = 2. This branch represents a Tactician and good for a Power Gamer; values of zero for the other types indicate that they are expected to be indifferent toward this branch, and negative values would indicate types for whom this branch should be avoided.

Leveraging Player Preferences

PaSSAGE creates its stories as a sequence of encounters drawn from a set of libraries; each library holds eight encounters based on three stages of the fairy tale “Little Red Cap” (Grimm & Grimm 1812). The encounter’s selection quality is then taken as the best quality value of its branches. The search for encounters can be initialized with a minimum desired quality or instructed to return the best encounter available.

Empirical Evaluation

In our previous publication, we evaluated PaSSAGE via a user study with respect to the following two hypotheses:

1. **Fun(A) > Fun(F):** Players feel that an adaptive story is more entertaining than a fixed story;
2. **Agency(A) > Agency(F):** Players feel more influential in an adaptive story than in a fixed story (Thue et al. 2007).

We present additional results in this section based on the same experimental setup as before. In brief, we used the Aurora Neverwinter Toolset (BioWare Corp. 2006) to create a library of eight encounters based on three stages of Campbell’s Momomyth: the Call to Adventure, Crossing the Threshold, and the Road of Trials. Each game began with a “history lesson” designed to allow PaSSAGE a chance to build a rough player model prior to choosing the first encounter, and the adaptive system (with PaSSAGE enabled) was tested alongside two stories with fixed plots chosen to collectively include every created encounter in variations of the fairy tale “Little Red Cap” (Grimm & Grimm 1812).

User Study Extension

To further explore our hypotheses, we extended our user study from 90 participants to 102. Each participant played through one of the three stories (two fixed and one adaptive), and filled out a post-game survey as before. Table 1 shows statistical significance results for our two hypotheses: (1) that players would find adaptive versions more entertaining than fixed stories, and (2) that players would feel higher agency in adaptive versions; the last two columns give confidence levels for their support. As before, the first two columns represent filters on the participants, designed to highlight segments of the population that might be well-targeted by a commercialization of our approach. A checkmark in the first column indicates that only players who noted having low previous gaming experience are considered (LE). A checkmark in the second column (ETF) limits participants to those who rated the game as being “easy to follow”. A blank in either column indicates no filtering. The columns labelled \( N_A \) and \( N_F \) list the number of filtered participants for the adaptive and fixed versions respectively.

For example, the first row (✓, ✓) shows that data from players with low prior gaming experience who found the game easy to follow support the hypothesis \( \text{Fun}(A) > \text{Fun}(F) \) with a confidence level of 91%. In other words, a T-test with a significance level of 9% (\( \alpha = 0.09 \)) rejects the null-hypothesis \( \text{Fun}(A) \leq \text{Fun}(F) \). As before, the last row (two blanks) deals with the data from all participants and fails to strongly support either of the hypotheses.

<table>
<thead>
<tr>
<th>LE ETF</th>
<th>( N_A )</th>
<th>( N_F )</th>
<th>( \text{Fun}(A) &gt; \text{Fun}(F) )</th>
<th>Agency(A) &gt; Agency(F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓ ✓</td>
<td>27 30</td>
<td>91%</td>
<td>87%</td>
<td></td>
</tr>
<tr>
<td>✓ ✓</td>
<td>36 35</td>
<td>80%</td>
<td>88%</td>
<td></td>
</tr>
<tr>
<td>✓ ✓</td>
<td>42 45</td>
<td>78%</td>
<td>73%</td>
<td></td>
</tr>
<tr>
<td>✓ ✓</td>
<td>51 51</td>
<td>65%</td>
<td>72%</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Confidence levels in support of our two hypotheses for four data subsets (LE = Low Experience, ETF = Easy To Follow).
Discussion

Our previous publication focused on PaSSAGE’s performance for female players; here we present data from players with low prior gaming experience. Interestingly, our results for the two common subsets (rows 3 and 4 in Table 1) have worsened by several percent. This occurrence may be explained by characteristics of the twelve participants who were added to extend our study: all had high previous gaming experience, and their ratings of PaSSAGE’s adaptations were generally low. Although our results for high-experience players are inconclusive, PaSSAGE’s poor performance for such players may be caused by the following effect: in most commercial story-based games, when an event occurs that is not directly caused by the player’s actions, she assumes (often correctly) that that event had no potential alternatives; this realization detracts from her sense of immersion and enjoyment of the game. Because PaSSAGE’s adaptations are driven indirectly through the player model, experienced players may have assumed (incorrectly) that each selected encounter was the only possible event, rating the game unfavourably as a result; in fact, every encounter had at least one alternative. Meanwhile, players unfamiliar with commercial interactive narratives may have envisioned a wider range of possible events, making them more appreciative when PaSSAGE chose encounters that were well-matched with their preferred style of play.

Future Work

In our efforts to advance the research of player modelling for interactive narratives, we have temporarily set several concerns aside. The most prominent concern (which is perhaps the most well-treated by others) is the question of what causes one encounter to be in any way causally linked with previous or subsequent encounters. Time constraints led us to solve this problem easily using a branching narrative tree structure, but as others in the field have pointed out, branching narratives are a poor solution at best (Mateas 2002; Magerko 2006). We have taken steps toward moving to a partial-order planning paradigm, and intend to do so as soon as a larger library of encounters has been made. The future work listed in our previous publication also remains.

Conclusion

This paper made the following three contributions. First, we presented a survey of related work in the rapidly growing field of player modelling in interactive narratives. Second, we introduced preference-informed delayed authoring: the task of automatically learning players’ preferences by observing their behaviour on-line and using those preferences to dynamically choose the events of an interactive narrative; we presented PaSSAGE as our implementation of this approach in the setting of interactive entertainment. Third, we extended our previous user study to 102 human participants, and found that compared to two fixed narratives, players with low prior gaming experience who found the game easy to follow felt higher agency in the adaptive version and rated adaptive gameplay as being more fun.

Acknowledgements

We gratefully thank all those who participated in our user study; their feedback was extremely helpful. We also appreciate support from both the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Informatics Circle of Research Excellence (iCORE).

References


