Working with Intelligent Narrative Technologies

David Thue

Abstract Artificial Intelligence systems have been used to generate narrative structures and simulate virtual story characters at a variety of different scales, across both academia and industry. Such systems are often built from specialized components known as intelligent narrative technologies. The goal of this chapter is to highlight some of the challenges that can arise when such technologies are used as part of authoring or executing an interactive story. Authoring in a way that works with these technologies often requires a host of technical skills, such as writing computer code, building mathematical models, or predicting the effect of a simple change on a large, complex system. In addition to explaining why these skills are needed and the problems that they help to solve, this chapter will highlight recent and ongoing efforts to make authoring for intelligent narrative technologies more accessible to those with fewer technical skills.

1 Intelligent Narrative Technologies

The phrase "intelligent narrative technologies" can have (at least) three meanings. One is that it describes a field of research, which studies how the techniques used by Artificial Intelligence (AI) systems can be applied in the context of narrative. While there are examples of such research from the early 1960s [42], it became more widespread in the 1990s [3, 15, 54, 22, 24] and continues actively to this day. Mateas and Sengers offer a detailed account of the early years of this research field in the first chapter of their book [25].

David Thue
RISE Research Group
School of Information Technology, Carleton University, Ottawa, Canada, and
Department of Computer Science, Reykjavik University, Reykjavik, Iceland
e-mail: david.thue@carleton.ca

This is a preprint of the following chapter: David Thue, Working with Intelligent Narrative Technologies, published in "The Authoring Problem: Challenges in Supporting Authoring for Interactive Digital Narratives", edited by Charlie Hargood, David Millard, Alex Mitchell, and Ulrike Spierling, 2022, Springer Cham, reproduced with permission of Springer Cham. The final authenticated version is available online at: http://dx.doi.org/[DOI forthcoming, pending publication]".

The second meaning of "intelligent narrative technologies" is that it is the name of a series of academic events, which began in 2007 [21] and was held most recently in 2020 [34]. In total, these events included two research symposia [20, 21], several workshops co-located with three academic research conferences [1, 2, 5, 14, 34, 52, 53, 55, 56], and a special track at the International Conference on Interactive Digital Storytelling [33]. For the past fifteen years, these events have been a common home for early-stage research done in the field of intelligent narrative technologies. A related series of events featured a recurring workshop on Computational Models of Narrative [13, 8, 9, 10, 11, 12, 32].

The third meaning of the phrase is more pragmatic, and it is the one that we focus on in this chapter: *intelligent narrative technologies* (INTs) are technologies that apply AI techniques in the context of narrative. They are the focus and products of the research done in the field of INT, and the primary topic of the papers that are published via the INT series of events.

What does it mean, then, to apply AI techniques in the context of narrative? Fundamentally, AI techniques can be applied to make decisions in an automated way, and working in narrative means making decisions in that context. In Interactive Digital Narrative (IDN), many decisions have been made using AI techniques, and these decisions have centered primarily on the potential *products* of an IDN system [16]. They answer questions that include (but are not limited to):

- *What characters and objects should exist in the narrative world?*
- *What should happen next in the story?*
- *What should this character do next?*
- *How should this character perform its next action or line of dialogue?*
- *How should the system respond to the player's last action?*

The methods that have been used to answer these questions are many and varied, and citations to works that explain some of them will appear throughout this chapter. The focus of this chapter, however, is different: rather than explain how a collection of INTs work, we aim to equip IDN authors with general strategies that might help them work more effectively with intelligent narrative technologies.

1.1 Authoring with a Narrative AI System

For the purposes of this chapter, we consider *authoring* to be a process of making and acting upon decisions about how some elements of a narrative (or perhaps many possible narratives) should be. This could involve creating characters, locales, key props, storyboards, and more. Furthermore, we consider a *narrative AI system* as a structured collection of one or more intelligent narrative technologies, each of which might apply different AI techniques; the system accepts one or more inputs and produces one or more outputs using the technology therein. For example, the AI-driven "drama manager" in *Façade* [28] accepts inputs including (i) a collection of dramatic beats (bundles of narrative content), (ii) an estimate of the story's current level of dramatic tension, and (iii) an author-desired trajectory of dramatic tension over time [23]. Given these inputs, it uses an optimization technique to identify a particular dramatic beat from the collection as its output: the one that best matches the next desired level of tension in the story.

It is common to say that an AI system "decides" which output(s) it should produce as it operates. Since authoring is about making and acting upon decisions, we say that an authoring process can *include* a narrative AI system; in such cases, the decisions that are made during authoring will be shared between the author(s) and the system. From an author's perspective, it can thus be useful to understand what drives a narrative AI system's decisions, along with how those decisions can be influenced. To build such an understanding, an author can pursue answers to the following key questions. We discuss strategies for tackling these questions as the core content of this chapter.

- How does the AI system behave?
- How can I influence the AI system's behaviour?
  - How can I determine the AI system's inputs?
  - What of the AI system itself can I change?
  - How can I refine or repurpose the AI system's outputs?

2 Understanding the Behaviour of a Narrative AI System

What can an author do to understand how a narrative AI system behaves? We discuss two types of strategy: experimentation and examination.

2.1 Experimenting with a Narrative AI System

This method of learning requires having access to the narrative AI system in a way that lets the author trigger and observe the results of system decisions under a variety of different circumstances. This might involve using support tools created by the system's developers; *Sentient Sketchbook* allows authors to see examples of the maps that it can generate and tune parameters that affect how they are generated [], while *Mimisbrunnur* allows authors to view potential sequences of narrative action that could occur during gameplay, given the content and constraints that the author has created [46]. It might also involve playtesting an IDN process that the narrative AI system influences, to bring the narrative world state into particular situations and observe how the AI system reacts. For example, much can be learned about *Façade*'s drama manager [28, 27] by playing *Façade* and observing how the AI system behaves.

In general, an author can produce different circumstances for a narrative AI system by modifying the system's inputs. Depending on what INTs the system uses, these inputs might include collections of various kinds of content, or parameters or utilities. We offer some examples of each.

2.1.1 Collections of Content as System Inputs

It is quite common for a narrative AI system to require one or more collections of content among its inputs, and the types of required content can vary widely across different systems.

As one example, several narrative AI systems that produce natural language text (*e.g.*, for character dialogue) require large collections of text to be provided as inputs; such collections might include film scripts, blog posts, news articles, and more. At a high level, such systems contain one or more INTs that perform natural language generation – they use the provided collections of text to build a general model of how people tend to write sentences and paragraphs in a given context (*e.g.*, in Science Fiction movie scripts), and then use the model to predict the words of new sentences and paragraphs. A compelling IDN system that uses this sort of technology is *AI Dungeon* [19].

Collections of images or 3D models are common inputs to narrative AI systems – particularly for those that are embedded in video games and generate some of their game's content. Procedural Content Generation (PCG) describes a process of automatically creating content (typically to be used in a game). When applied to narrative contexts, PCG methods become INTs, as they are used to make decisions about how narratively-important content should be. Examples include the generation of non-player characters (including their appearance and attributes) [31], towns [44], and more. Two compelling IDN systems that use PCG to create narrative content are *Dwarf Fortress* [4] and *Rimworld* [29].

A frequently studied subtopic of INT research is Narrative Planning [41], which uses automatic, logical reasoning to find plans of action for story characters that satisfy goals given by an author. To form such plans, the narrative planner (which is a narrative AI system) requires a collection of potential actions for characters to perform, plus a collection of characters and other entities (*e.g.*, props and locations) whose attributes can be changed by the given actions. For example, the authoring tool *Mimisbrunnur* allowed authors to create collections of actions and entities, and then preview examples of how a narrative planner might use those actions to achieve different story goals [46].

By adjusting the collections that a narrative AI system receives in its inputs, an author can put the system in different circumstances and observe how it behaves therein.

2.1.2 Parameters & Utilities as System Inputs

Many narrative AI systems have *parameters* – variables that are meant to be adjusted by the user of the system to alter its behaviour. For example, at the beginning of a game of *Rimworld*, players are able to set a variety of parameters that control how the game's narrative AI system will behave [29]. The system generates notable events from a library of templates (*e.g.*, attacks by hostile creatures or extreme weather), and the parameters affect various aspects of how those events get generated (*e.g.*, their frequency or severity).

One weakness of parameters is that they each remain fixed at their given value, regardless of what might happen while the system is operating. When it is important for a variable's value to change in response to changing circumstances, a narrative AI system might require an input that helps it compute new values for that variable. This sort of input can be well thought of as a *utility* – something that allows new values of a variable to be calculated given other values (*e.g.*, of other variables, or of attributes of entities in the narrative world). For example, the drama manager in *Façade* [28] requires a utility that does the following: It starts by considering the history of the story's events thus far, and retrieves estimates of how each event should have contributed to the story's dramatic tension. Then, by adding these contributions together, the utility produces an estimate of the *current* state of dramatic tension in the story. This utility is used regularly during gameplay to estimate the story's current tension level [23].

By adjusting the parameters or modifying the utilities that a narrative AI system uses, an author can create different circumstances for the AI system and observe how it behaves as a result.

2.2 Examining a Narrative AI System

Beyond observing a narrative AI system to gain an impression of how it behaves, authors who are more technically inclined may be able to examine the system itself, toward learning how its internal mechanisms lead it to behave in different ways. Such an examination might involve reading publications or other technical documents, or reading the program code that executes while the system operates.

When a narrative AI system is described across multiple publications of different types and lengths, it can be challenging to know where to start. Research papers published at academic workshops or conferences, as well as white papers and blog posts written by developers, typically contain high level explanations in a relatively compact presentation. These works can be useful for identifying the required inputs of a given system, but many also contain a concise description of how the system works, either through figures, pseudocode, or both. Writing in pseudocode allows a system developer to express the key steps of the system's program code in a way that is more readable than the program code itself, and pseudocode should ideally be understandable even with only a limited knowledge of computer programming.

As an example, Algorithm 1 shows pseudocode that explains the operation of a kind of narrative AI system called a *player-specific experience manager*[1] [49]. The

[1] The term *experience manager* is due to Riedl *et al.* [40], and refers to an AI system that attempts to modify the course of a player's experience as it proceeds.

Algorithm 1: Pseudocode showing the high-level operation of a narrative AI system that adapts a story using a learned player model. Italics show variables and upright text shows utilities that compute useful values. The notation "$x \leftarrow y$" means "x gets set to the value of y".

```
Inputs  : narrativeWorldState: the current state of the narrative world
          playerAction: the most recent action that the player performed
          currentPlayerModel: prior information learned about the player
          possibleAdaptations: a collection of ways to adapt the current story
Outputs: nextAdaptation: the adaptation that best matches with the player model

1 currentPlayerModel ←
    GetUpdatedPlayerModel(currentPlayerModel, narrativeWorldState, playerAction)
2 bestMatchQuality ← 0
3 for each adaptation in possibleAdaptations do
4     matchQuality ← EstimateMatchQuality(adaptation, currentPlayerModel)
5     if matchQuality > bestMatchQuality then
6         bestMatchQuality ← matchQuality
7         nextAdaptation ← adaptation
8 return nextAdaptation
```

inputs and expected output are stated at the top. On Line 1, a current model of the player (*e.g.*, representing their preferences) is updated based on the current narrative world state and the player's most recent action. On Lines 2 to 7, the set of possible adaptations is searched, element by element, while estimating how well each possible adaptation matches with the current player model (Line 4). Each time a better match is found, it is set as the next adaptation that should be performed to adapt the current story (Line 7). The result is that the adaptation that matches the player model the best is the one that will be output. Although this example has been simplified for the sake of introducing the concept (notably by avoiding most mathematical notation), it nonetheless demonstrates the rough character of how pseudocode is presented in technical writing about narrative AI systems.

Given the ability to read its pseudocode or program code, an author can directly examine the operation of a narrative AI system, and use what they discover to understand the system's general behaviour.

3 Ways to Influence a Narrative AI System

Once an author has come to understand how a narrative AI system might behave, their attention might turn to the question of how they can make it behave *differently*. In this section, we consider the remainder of the questions that we asked in Section 1.1, including how an author might determine the inputs of a narrative AI system, how they might alter the system itself, and how they might refine or repurpose what the system produces as output.

3.1 Determining the System's Inputs

As we discussed in Section 2.1, it is often possible for an author to change various inputs of a narrative AI system. Indeed, the majority of the inputs of many narrative AI systems are expected to be authored by one or more people. For example, *Rimworld*'s world generator (another narrative AI system) can generate an entire unique planet, including terrain, biomes, creatures, settlements, and inhabitants. From a player's perspective, it seems to generate all of this from very little: a single starting seed (a random string of characters) and a handful of generation parameters. In reality, the world generator works by cleverly combining many collections of pre-authored content, including trees, boulders, creatures, building materials, props, character attributes, character appearances, and more. The starting seed and generation parameters matter, but the pre-authored content provides the bulk of the resources that are used while the generator works. As a result, one way to influence how *Rimworld*'s world generator behaves could be to modify one or more of these collections of content, all of which the generator uses as inputs. Other ways include changing either the starting seed or the generation parameters.

An important difference can be seen between these methods of exerting influence: some are more *controllable* than others, in that the outcome of any change is easier to predict and thus easier to use in intentional ways. While the effects of replacing a boulder in the pre-authored content seems relatively easy to predict and thus more straightforward to control, the effects of changing a starting seed are nearly impossible to predict. This makes the seed value more challenging to use when pursuing particular authorial aims.

Beyond parameters and collections of content, it is also often expected that the utilities that are required by a narrative AI system will be authored by one or more people. For example, the trajectory of dramatic tension over time that *Façade*'s drama manager requires is explained as being provided by an author [27, 28]. Utilities offer a way to influence the way that a narrative AI system will behave, because they are often used by such systems to (i) differentiate between potential alternatives and (ii) infer useful meaning from the narrative world state. *Façade*'s desired tension trajectory is an example of the former, while its method for estimating the story's *current* level of dramatic tension is an example of the latter.

In player-specific experience management (recall Algorithm 1), one or more utilities are used to estimate a player model [51, 45, 50, 49]. A *player model* is a mathematical representation of some aspect of a player; this might represent their personality, their knowledge of the story, their preferences over different types of content, their expected emotional reactions, or more. By defining the dimensions of a player model and creating a utility to estimate each of them, an author can influence how the experience manager makes its decisions.

3.2 Altering the System

If an author can gain access to the program code that defines how a narrative AI system works, they might be able to change it to make it work in a different way. In one example, Riedl & Stern took the *ABL* behaviour specification language that was created for *Façade* [28, 26] and combined it with an experience manager based on Narrative Planning called the *Automated Story Director* [39]. Later, Ramirez and Bulitko obtained the source code to the *Automated Story Director* and adapted it to add a player model based on Thue *et al.*'s *PaSSAGE* [51], creating a player-specific experience manager called *PAST* [36, 35].

Notably, all of the prior works are examples of technically-savvy people acting both as the developers of narrative AI systems *and* as the authors of the IDNs that included those systems. While positive examples exist of diversely skilled teams creating compelling IDNs that rely on narrative AI systems (including *Prom Week* [31, 30], *The Ice-Bound Concordance* [38, 37], *Blood & Laurels* [7, 6], *Nothing for Dinner* [48, 47], and more), finding a way to make system modification more widely accessible to authors remains an open research problem.

With *Mimisbrunnur*, authors were able to preview outputs of its narrative planner and, if desired, mark any outputted plan of action as an unacceptable solution [46]. From that point forward, the system would remember and abide by that decision, never showing the marked solution again. While simple in this application, adding a similar capacity for incremental modification to future narrative AI systems might allow authors to alter their operation in a more accessible way.

3.3 Refining or Repurposing a System's Outputs

Throughout this chapter, we have considered authoring as a process of making and acting upon decisions in a narrative context. While narrative AI systems have been used to make a variety of authoring decisions, the task of acting upon those decisions (*e.g.*, making the next dramatic beat actually happen in *Façade*) it is typically left to other parts of the IDN system. This interface that exists between the narrative AI system's output and the remainder of the IDN system presents the last opportunity for an author to influence the decisions that the narrative AI system makes: if each decision can be intercepted and revised or repurposed as desired, the author can gain the benefit of the AI system's operation while still influencing its results. The mode in which an author can do this sort of refinement or repurposing depends on the timing of the narrative AI system's decisions, relative to any player's experience of the larger IDN system's product.

For AI system decisions that are made *before* any player's experience (*e.g.*, to generate a backstory for a character that every player will encounter), choosing among several of the system's outputs by hand can be a viable option – provided that the system can generate outputs quickly enough to be useful. In this mode of refinement, the system's output serves a starting point for the author's subsequent creative process. When the size of the set of possible outputs is very large, a degree of automation can help. *Story sifting* [43, 18] is a process of searching through a generated sequence of events and identifying subsequences that are salient in some way, and this is typically done on the basis of flexible patterns that one or more authors specify to guide the search.

When AI system decisions are made *during* any player's experience, a more nimble approach is required. For example, given an interactive narrative AI system that simulated the social interactions of several story characters, an author might create one or more utilities that attempt to recognize certain patterns of happenings (*e.g.*, those that involve one character betraying another) and bring the matches to the player's attention. This process is called *incremental story sifting* [17], and it represents some of the newest work in this direction.

Summary

From an author's perspective, intelligent narrative technologies are the elements of narrative AI systems, and these systems can share in the task of authoring IDN systems and products. When working with a narrative AI system, it can benefit an author to understand both how the system behaves and what they might do to influence its behaviour. This behaviour can be understood by either experimenting with the system to observe it under different circumstances, or examining its pseudocode or program code to learn about how it works. Meanwhile, an author's opportunities to influence a narrative AI system's decisions come in three forms. First, they can determine its inputs, either by building the collections of content that it uses, settings its parameters, or defining the utilities that it uses as a part of its operation. Second, they can modify how the system itself works by editing its program code, though this avenue presently lacks accessibility for non-technical authors. Third, they can refine or repurpose the output that the AI system produces, potentially by specifying patterns that identify outputs that are of particularly high value. By understanding how a narrative AI system behaves and what they can do to make it better, authors can benefit from the generative capabilities of AI systems while still pursuing the stories they wish to tell.

References

1. Bahamón, J., Fassone, R., Mawhorter, P., Poulakos, S., Robertson, J., Ryan, J. (eds.): Intelligent Narrative Technologies - Papers from the 2017 AIIDE Workshop, vol. WS-17-20. AAAI Press (2017)
2. Barot, C., Li, B., Rowe, J., Tomai, E. (eds.): Intelligent Narrative Technologies and Social Believability in Games: Papers from the 2015 AIIDE Workshop, vol. WS-15-22. AAAI Press (2015)

3. Bates, J.: Virtual reality, art, and entertainment. Presence: The Journal of Teleoperators and Virtual Environments **1**(1), 133–138 (1992)
4. Bay 12 Games: Dwarf Fortress. http://www.bay12games.com/dwarves/ (2006–2021)
5. Cavazza, M., Si, M., Zook, A. (eds.): Intelligent Narrative Technologies: Papers from the 2013 AIIDE Workshop, vol. WS-13-21. AAAI Press (2013)
6. Evans, R., Short, E.: Versu—a simulationist storytelling system. IEEE Transactions on Computational Intelligence and AI in Games **6**(2), 113–130 (2014). DOI 10.1109/TCIAIG.2013.2287297
7. Evans, R., Short, E., Nelson, G.: Blood & Laurels. https://versu.com/2014/05/28/blood-laurels/ (2014)
8. Finlayson, M.A. (ed.): Computational Models of Narrative - Papers from the 2010 AAAI Fall Symposium, vol. FS-10-04. AAAI Press (2010)
9. Finlayson, M.A. (ed.): The Third Workshop on Computational Models of Narrative (CMN'12). narrative.csail.mit.edu (2012)
10. Finlayson, M.A., Fisseni, B., Löwe, B., Meister, J.C. (eds.): 2013 Workshop on Computational Models of Narrative (CMN 2013), vol. 32. OASICS (2013)
11. Finlayson, M.A., Meister, J.C., Bruneau, E.G. (eds.): 2014 Workshop on Computational Models of Narrative (CMN 2014), vol. 41. OASICS (2014)
12. Finlayson, M.A., Miller, B., Lieto, A., Ronfard, R. (eds.): 6th Workshop on Computational Models of Narrative (CMN 2015), vol. 45. OASICS (2015)
13. Finlayson, M.A., Richards, W., Winston, P.H.: Computational models of narrative: Review of a workshop. AI Magazine **31**(2), 97 (2010). DOI 10.1609/aimag.v31i2.2295. URL https://ojs.aaai.org/index.php/aimagazine/article/view/2295
14. Jhala, A., Riedl, M., Roberts, D.L. (eds.): INT3 '10: Proceedings of the Intelligent Narrative Technologies III Workshop. Association for Computing Machinery, New York, NY, USA (2010)
15. Kelso, M.T., Weyhrauch, P., Bates, J.: Dramatic presence. Presence: The Journal of Teleoperators and Virtual Environments **2**(1), 1–16 (1993)
16. Koenitz, H.: Towards a theoretical framework for interactive digital narrative. In: R. Aylett, M.Y. Lim, S. Louchart, P. Petta, M. Riedl (eds.) Interactive Storytelling, pp. 176–185. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
17. Kreminski, M., Dickinson, M., Mateas, M.: Winnow: A domain-specific language for incremental story sifting. In: Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, pp. 156–163. AAAI Press (2021). URL https://ojs.aaai.org/index.php/AIIDE/article/view/18903
18. Kreminski, M., Dickinson, M., Wardrip-Fruin, N.: Felt: A simple story sifter. In: Proceedings of the 12th International Conference on Interactive Digital Storytelling (ICIDS 2019), pp. 267–281. Springer-Verlag, Berlin, Heidelberg (2019)
19. Latitude: AI Dungeon. https://play.aidungeon.io/ (2019)
20. Louchart, S., Mehta, M., Roberts, D.L. (eds.): Intelligent Narrative Technologies II - Papers from the 2009 AAAI Spring Symposium, vol. SS-09-06. AAAI Press (2009)
21. Magerko, B.S., Riedl, M.O. (eds.): Intelligent Narrative Technologies - Papers from the 2007 AAAI Fall Symposium. AAAI Press (2007)
22. Mateas, M.: An oz-centric review of interactive drama and believable agents. Tech. rep., Carnegie Mellon University (1997)
23. Mateas, M.: Interactive drama, art, and artificial intelligence. Ph.D. thesis, Carnegie Mellon University (2002)
24. Mateas, M., Sengers, P. (eds.): Narrative Intelligence - Papers from the 1999 Fall Symposium, vol. FS-99-01. AAAI Press (1999)
25. Mateas, M., Sengers, P. (eds.): Narrative Intelligence. John Benjamins (2003). URL https://www.jbe-platform.com/content/books/9789027297068
26. Mateas, M., Stern, A.: A behavior language: Joint action and behavioral idioms. In: H. Prendinger, M. Ishizuka (eds.) Life-Like Characters: Tools, Affective Functions, and Applications, pp. 135–161. Springer Berlin Heidelberg, Berlin, Heidelberg (2004)

27. Mateas, M., Stern, A.: Structuring content in the Façade interactive drama architecture. In: Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, pp. 93–98. AAAI Press (2005). URL https://ojs.aaai.org/index.php/AIIDE/article/view/18722
28. Mateas, M., Stern, A.: Façade. http://www.interactivestory.net/ (2007)
29. Ludeon Studios: Rimworld. https://rimworldgame.com (2013)
30. McCoy, J., Treanor, M., Samuel, B., Reed, A., Mateas, M., Wardrip-Fruin, N.: Prom week: Designing past the game/story dilemma. In: Proceedings of the 8th International Conference on the Foundations of Digital Games (FDG 2013). Chania, Crete, Greece (2013)
31. McCoy, J., Treanor, M., Samuel, B., Reed, A.A., Mateas, M., Fruin, N.W.: Prom Week. https://promweek.soe.ucsc.edu/ (2012)
32. Miller, B., Lieto, A., Ronfard, R., Ware, S.G., Finlayson, M.A. (eds.): 7th Workshop on Computational Models of Narrative (CMN 2016), vol. 53. OASICS (2016)
33. Nack, F., Gordon, A.S. (eds.): Interactive Storytelling: 9th International Conference on Interactive Digital Storytelling (ICIDS 2016), vol. LNCS 10045. Springer, Cham (2016)
34. Osborn, J.C. (ed.): Joint Proceedings of the AIIDE 2020 Workshops, co-located with 16th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2020), vol. 2862. CEUR Workshop Proceedings (2020). URL http://ceur-ws.org/Vol-2862/
35. Ramirez, A., Bulitko, V.: Automated planning and player modelling for interactive storytelling. IEEE Transactions on Computational Intelligence and AI in Games **7**(4), 375–386 (2015)
36. Ramirez, A.J., Bulitko, V.: Telling interactive player-specific stories and planning for it: ASD + PaSSAGE = PAST. In: The Eighth Artificial Intelligence and Interactive Digital Entertainment Conference (AIIDE 2012), pp. 173–178. AAAI Press, Palo Alto, California (2012)
37. Reed, A.: Changeful tales: Design-driven approaches toward more expressive storygames. Ph.D. thesis, UC Santa Cruz (2017)
38. Reed, A., Garbe, J., Apostol, N.: The Ice-Bound Concordance. www.ice-bound.com (2013)
39. Riedl, M.O., Stern, A.: Believable agents and intelligent story adaptation for interactive storytelling. In: 3rd International Conference on Technologies for Interactive Digital Storytelling and Entertainment, pp. 1–12. Springer, Darmstad, DE (2006)
40. Riedl, M.O., Stern, A., Dini, D., Alderman, J.: Dynamic experience management in virtual worlds for entertainment, education, and training. International Transactions on Systems Science and Applications, Special Issue on Agent Based Systems for Human Learning **4**(2), 23–42 (2008)
41. Riedl, M.O., Young, R.M.: Narrative planning: Balancing plot and character. Journal of Artificial Intelligence Research **39**, 217–268 (2010)
42. Ryan, J.: Grimes' fairy tales: A 1960s story generator. In: N. Nunes, I. Oakley, V. Nisi (eds.) Interactive Storytelling, pp. 89–103. Springer International Publishing, Cham (2017)
43. Ryan, J.: Curating simulated storyworlds. Ph.D. thesis, University of California, Santa Cruz (2018)
44. Salge, C., Green, M.C., Canaan, R., Togelius, J.: Generative design in minecraft (gdmc): Settlement generation competition. In: Proceedings of the 13th International Conference on the Foundations of Digital Games, FDG '18. Association for Computing Machinery, New York, NY, USA (2018). DOI 10.1145/3235765.3235814. URL https://doi.org/10.1145/3235765.3235814
45. Sharma, M., Mehta, M., Ontañón, S., Ram, A.: Player modeling evaluation for interactive fiction. Tech. rep., AIIDE 2007 Workshop on Optimizing Player Satisfaction, Palo Alto, California: AAAI Press (2007)
46. Stefnisson, I., Thue, D.: Mimisbrunnur: AI-assisted authoring for interactive storytelling. In: Proceedings of the 14th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, pp. 236–242. AAAI Press (2018). URL https://aaai.org/ocs/index.php/AIIDE/AIIDE18/paper/view/18116/17248
47. Szilas, N.: Reconsidering the role of ai in interactive digital narrative. In: H. Koenitz, G. Ferri, M. Haahr, D. Sezen, T.İ. Sezen (eds.) Interactive Digital Narrative. Routledge (2015)
48. Szilas, N., Dumas, J., Richle, U., Boggini, T., Habonneau, N.: Nothing for dinner. http://nothingfordinner.org/ (2015)

49. Thue, D.: Generalized experience management. Ph.D. thesis, University of Alberta, Canada (2015)
50. Thue, D., Bulitko, V., Spetch, M.: Making stories player-specific: Delayed authoring in interactive storytelling. In: The First Joint International Conference on Interactive Digital Storytelling, pp. 230–241. Springer Berlin / Heidelberg, Erfurt, Germany (2008)
51. Thue, D., Bulitko, V., Spetch, M., Wasylishen, E.: Interactive storytelling: A player modelling approach. In: 3rd AI and Interactive Digital Entertainment Conference (AIIDE 2007), pp. 43–48. AAAI Press, Palo Alto, California (2007)
52. Tomai, E., Elson, D., Rowe, J. (eds.): Intelligent Narrative Technologies IV - Papers from the 2011 AIIDE Workshop, vol. WS-11-18. AAAI Press (2011)
53. Ware, S.G., Zhu, J., Hodhod, R. (eds.): Intelligent Narrative Technologies - Papers from the 2012 AIIDE Workshop, vol. WS-12-14. AAAI Press (2012)
54. Weyhrauch, P.: Guiding interactive drama. Ph.D. thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA (1997)
55. Wu, H.Y., Si, M., Jhala, A. (eds.): Proceedings of the Joint Workshop on Intelligent Narrative Technologies and Workshop on Intelligent Cinematography and Editing co-located with 14th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2018), vol. 2321. CEUR Workshop Proceedings (2018)
56. Zhu, J., Horswill, I., Wardrip-Fruin, N. (eds.): Intelligent Narrative Technologies 7 - Papers from the 2014 Workshop, vol. WS-14-21. AAAI Press (2014)