Sharing Authoring with Algorithms

Procedural Generation of Satellite Sentences in Text-based Interactive Stories

Aaron A. Reed
UC Santa Cruz
1156 High Street
Santa Cruz, CA 95064
aareed@soe.ucsc.edu

ABSTRACT

As both a fiction writer and a computer scientist, I want the interactive stories I create to be *meaningfully* interactive: choices should matter. To avoid laborious hand-authoring of variations, procedural content generation (PCG) seems appealing; but PCG has been less successful in producing compelling narrative text than in other realms. To address this problem, I consider the minimum amount of PCG that might make a human-authored story computationally interesting but still authorially sound. The resulting research prototype generates "satellite" sentences (which moderate pacing and reestablish context within dialogue scenes) within otherwise hand-authored scenes in a complete interactive story.

1. INTRODUCTION

Procedural content generation (PCG) can be a frustrating collaborator. Artists and authors have specific visions for the stories they wish to tell, which are not always easy to share with the algorithms driving PCG systems. While PCG has been successfully used to assist with aspects of many released games, from *Rogue* to *Minecraft*, such generation is rarely used to affect the narrative (if one is even present). The few exceptions tend to prove the accepted rule: that procedurally generated narrative cannot yet compete with hand-authored stories in the same way that (for instance) procedural terrain can come close to emulating real-world or hand-crafted terrain. The best procedurally generated texts to date are successfully functional, but do not yet win writing awards, nor are they generally read for pleasure.

As an author of interactive stories, I want my work to be beautiful, not merely functional. I also want stories that are meaningfully interactive: choices should matter. The typical compromise to satisfy both constraints is a laborious hand-authoring of every possible story state. But the combinatorial explosion that results creates an untenable authorial burden. As a computer scientist, then, my research question is this: How can authors maintain quality control over interactive stories, while still allowing them to respond to player choice, without the burden of hand-authoring every possible consequence of those choices? How can a writer find a PCG collaborator he can trust not to stick out like a sore thumb?

As a step towards addressing this question, I began with an entirely human-generated story, and considered the minimum amount of procedural generation that might vary it in a nontrivial way while still producing text as good as if I had hand-authored it. The resulting research prototype generates *satellite* sentences (which moderate pacing and reestablish context) which are inserted into an otherwise hand-authored interactive story, and is the primary contribution of this paper. Generation is accomplished with an adaptive set of grammars configured based on the current narrative context, which includes the consequences of past player choices, tracking of the current speaker, time, and place. The resulting story, "Almost Goodbye," aspires to be both computationally interesting and narratively sound, casting the player as an interstellar colonist, about to leave Earth forever, saying her final goodbyes. Readers can play the story for themselves at almostgoodbye.textories.com; it takes about ten minutes.

2. RELATED WORK

The aims of this project are more specific than the broader problem of generating narrative text, which has been addressed before in several contexts. Generation of complete narratives from scratch has been tried in a number of different frameworks such as Tale-Spin[7], UNIVERSE[6], and MEXICA[9] although the artifacts produced by these systems rarely stand alone as compelling prose. Similarly, generation of text to describe a simulated story world (as in the GLINDA[5] or AUTHOR[2] systems) has successfully produced functional but rarely noteworthy sentences. In both cases the system is responsible for producing most of the text from scratch, a much more difficult problem than collaborating with a capable human author. Conversely, the Curveship[8] system also describes a simulated story world, but relies almost entirely on human-authored text, varying only the narration style by altering manually tagged verbs, subjects, and objects. The system under discussion here aims for a middle road, relying mostly on human authoring but turning some content over to a generator.

PCG has been employed more frequently for digital poetry and narrative toys. Reducing the need to tell a coherent story relaxes the constraints on such systems considerably, since unexpected or imperfect output can more easily be forgiven or read as intentional. One complex example is GRIOT[4], which generates author-driven poetic fragments using a theory of conceptual blending to create emergent metaphorical connections. Narrative toys similarly explore fragments of narrative rather than complete stories, as in the iPad app "My Secret Hideout"[10] which creates text describing a treehouse that is continuously varied as the user adds, repositions, and removes nodes from an onscreen tree of symbols. The paper author has also previously produced work in this vein, such as "Perfect,"[11] where dragging colored blocks around a screen alters the text of a five-line story to produce a narrative fragment which is sometimes coherent but often not. While these systems all produce narratively-inspired artifacts, they are not stories in the literary sense with qualities such as a narrative arc, and a beginning, middle, and end.

Procedural generation of quests[3, 13] and puzzles[1] has also been a target of prior PCG games research. Quests in particular are expected to provide narrative content, although this is nearly always self-contained: the player's choices within a quest, even if any are offered, are rarely remembered past its solution. A quest or puzzle is conceptualized as an interlude in a larger story, but it offers few affordances for the player to affect that containing story. This project focuses on the challenges of variation and responsiveness within a complete story rather than a self-contained fragment.

3. DESCRIPTION OF WORK

For this project I created a system to support author-driven text-based interactive stories by handling runtime generation of two types of sentences which might appear during dialogue scenes, *anchoring* and *pacing* sentences. The system built to generate these sentences (Figure 1) consists of three major parts, described in detail below: 1) a library of **authored content**, including dialogue scenes and context-specific expansion grammars; 2) a **grammar parser** that selects a set of grammars from the library based on the current narrative context; and 3) a **realizer** that expands the dialogue scenes with generated sentences, displays the results, and allows the player to make choices that change the narrative context, such as by altering the set of descriptors for the main character, altering the available grammars.

Generation Domain. In fiction, *anchoring sentences* remind the reader of the situational context a scene takes place in, and *pacing sentences* add pauses or performative spacing between moments of spoken text. Examples of both types from some popular novels appear in Table 1. These types of sentences can be called *satellite* rather than *kernel* sentences because rather than directly narrating the events of a story they "fill in the outline of a sequence by maintaining, retarding, or prolonging the kernel events they accompany or surround"[12]. These two types of satellite sentences were chosen for generation because 1) they tend to be short and simple, 2) they are only loosely coupled to the surrounding narrative, and thus need to understand less about it than other types of sentences, and 3) they are amenable to PCG techniques, with many possible variations on a small subset of possible English sentences. Furthermore, as these types of sentences tend to be sprinkled throughout a scene, 4) they allow generated text to be interwoven with authored text, letting the player-influenced context be referenced repeatedly amidst the context-blind pre-authored text.

Anchoring sentences
In the hot sunlit pastures yellow flowers bloomed.
It was getting late.
The day was waning, and the muttering of the city continued outside.
The scrape of a chair.
Pacing sentences
There was a silence.
He smiled again.
She let that sink in.
He took a deep breath.
The tension was unbearable.

Table 1: Examples of human-authored sentences in the generation domain taken from novels.

1. Authored Content. The format for authoring both grammars and scenes was deliberately made as simple as possible to allow the writer to focus on writing and not worry about syntax. For scenes, the author writes a set of conversations, each (for "Goodbye") between the narrator and one other character. Moments where a satellite sentence would be appropriate are indicated with a percent sign (%). The author also indicates when quoted text begins referring to a new speaker with an underscore and a unique number for each character.

```
LOCATION: *DINERBUSINESS - *LOCATIONACTORBUSINESS
DINERBUSINESS: *DINERBUSTLE - *DINERLIGHTONHIM -
*DINERSMELL
DINERBUSTLE: *DINERPATRON *OPTLOCATION *DINERVERB
DINERPATRON: Someone - a waitress - a woman - a diner
DINERVERB: sneezes - spills a coffee and swears -
clinks a plate - drops a fork - laughs uproariously -
drops a fork on the tiled floor - is arguing over the
check - talks noisily on her phone
DINERLIGHTONHIM: *LIGHT *SHINES on *HIMHER
```

Table 2: Some expansion rules from a location grammar for a diner, one possible place a scene from "Almost Goodbye" might take place.

The author also writes a set of short grammars in a simplified format (Table 2) that generate different kinds of satellite sentences in each context necessary for the current story. For "Goodbye," the relevant contexts are location, time of day, narrator descriptors, current conversant, and two universal grammars (for pauses and pacing). "Goodbye" features five possible locations, five conversants, four times of day, and seven narrator descriptors, so 23 total short grammars were written including the two universal grammars.

To encourage personalization within individual sentences, each category of grammar can specify that members must contain certain expansion rules, which can be referenced by any other grammar. For example, the time of day category requires any member to contain an expansion rule called *LIGHT, which might expand to "hot summer light" in the *Afternoon* grammar or "moonlight" in the *Midnight* grammar. This allows single generated sentences to incorporate context from several sources, allowing for sentences like "The

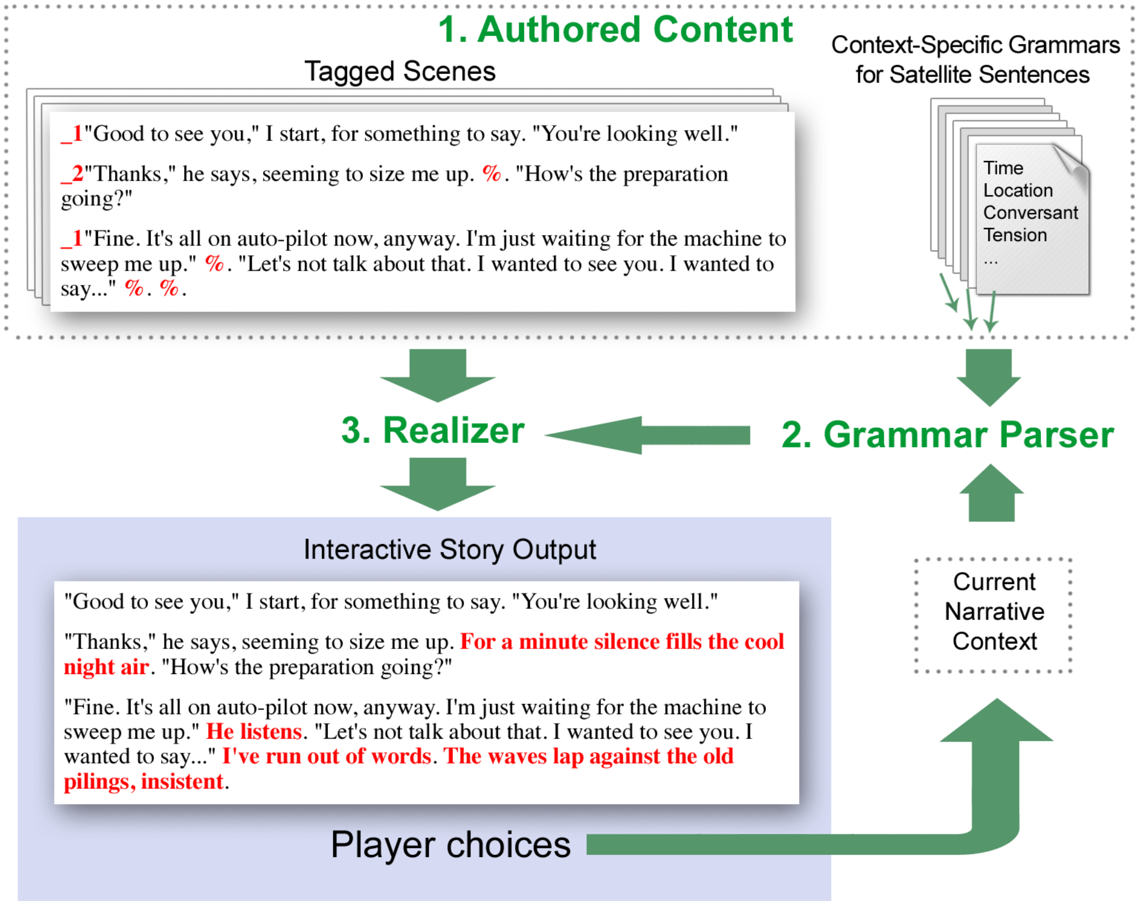
1. Authored Content
Tagged Scenes
_1"Good to see you," I start, for something to say. "You're looking well."
_2"Thanks," he says, seeming to size me up. %. "How's the preparation going?"
_1"Fine. It's all on auto-pilot now, anyway. I'm just waiting for the machine to sweep me up." %. "Let's not talk about that. I wanted to see you. I wanted to say..." %. %.
Context-Specific Grammars for Satellite Sentences
Time
Location
Conversant
Tension
...
3. Realizer
2. Grammar Parser
Interactive Story Output
"Good to see you," I start, for something to say. "You're looking well."
"Thanks," he says, seeming to size me up. For a minute silence fills the cool night air. "How's the preparation going?"
"Fine. It's all on auto-pilot now, anyway. I'm just waiting for the machine to sweep me up." He listens. "Let's not talk about that. I wanted to see you. I wanted to say..." I've run out of words. The waves lap against the old pilings, insistent.
Player choices
Current Narrative Context

Figure 1: Architecture for the prototype system.

moonlight sparkled on him as he sipped his coffee," where *moonlight*, the male pronouns, and *sipped [his] coffee* each come from a different grammar.

2. Grammar Parser. The parser takes the current narrative context and selects, from the set of possible grammars, which to load for each particular scene. The selected grammars are merged into a single compound grammar for this narrative context, which always includes some static parameters (such as the universal grammar for narrating moments of silence), some which change without player intervention (such as the advancing time of day), and some which are responsive to player input (the character and location are selected by the player, and the narrator has a set of descriptors that can be gained or lost based on player choice in each conversation). The relevant text files containing the authored grammars are loaded and parsed into an executable form.

3. Realizer. The realizer takes the tagged scene and the combined grammar and fills in the tags with generated satellite sentences (Table 3). The realizer considers both the author's top-level weighting of the different grammar categories and a tag's position relative to the surrounding dialogue. This position can be in one of four categories: (a) a new speaker is just about to start speaking, (b) a speaker has just finished speaking, (c) a speaker is pausing and is about to continue, or (d) no recognized dialogue position. Along with information about the identities of the former and next speaker, this allows the dialogue pacing universal grammar to seem aware of the dialogue context, producing sentences like "She licked her lips before continuing" or "I stopped, waiting for his reply."

To avoid repetition which might reveal the presence of automation, the system keeps a list of nontrivial words that have appeared in generated sentences so far. Any subsequent generated sentences containing one of these words are re-generated. This mimics a human author's tendency to avoid repeating words in close proximity to each other, and also encourages content variation in the satellite sentences.

Once all tags are filled with realized text, the scene is displayed to the player. To add a simple amount of interactivity in "Goodbye," each scene lets the player make one choice between two options. Each choice involves the nature of the player-character's relationship with the conversant: a mother figure, for instance, can be obeyed or overruled. Either decision affects a list of adjectives describing the narrator, which is visible to the player, by adding some and removing others. A player who begins a scene with the descriptors *Afraid*, *Conflicted*, and *Driven* might end it with *Afraid*, *Sure*, and *Driven* (having lost *Conflicted* but gained *Sure*). Each descriptor is associated with its own grammar.

We'll get through this. *(Sure)*
My head pounds furiously. *(Driven)*
The sun warms me. *(Afternoon)*
I feel desperate to grab hold of something. *(Afraid)*
I sigh. *(Dialogue)*
She listens. *(Dialogue, Conversant)*
There's still time. *(Afternoon)*
I tap my fingers rhythmically, tap, tap, tap. *(Driven)*
The summer sunlight casts sharp shadows behind the waves. *(Docks, Afternoon)*
I lick my lips. *(Dialogue)*
Maybe I shouldn't be going. *(Afraid)*
I stare at the peeling paint on the boards at my feet. *(Docks)*
There's no time for this. *(Driven)*
I take a deep breath. *(Dialogue)*
I consider my words carefully. *(Dialogue)*
She hesitates. *(Dialogue, Conversant)*
A seagull cries brokenly from somewhere above, lost. *(Docks)*
The low blast of a tug horn rolls over the waves. *(Docks)*

Table 3: Sample generated sentences from "Goodbye," and the grammar responsible for each.

Therefore the player's choices of how the main character acts in her final moments subtly affect the satellite sentences in the following scenes. A *Conflicted* player-character might be described as agonizing over a long silence from a conversant, whereas a *Sure* character might confidently wait for a reply.

The player also selects the order and location of each conversation. In "Goodbye" these are mostly trivial choices, except that the piece is structured such that the player does not have enough time to say goodbye to all five characters: one of them will inevitably be left out, which narratively frames the static conclusion of the piece.

4. CONCLUSIONS AND FUTURE WORK

The process of crafting a complete story with this system led to a number of interesting insights. Originally, I expected to want more control over the specifications of each request for a generated satellite sentence, envisioning the need for tag markers to request sentences of a certain length, subject matter, or style. In practice, a simple unqualified tag tended to produce satisfactory results in most cases: in a sample traversal of the story, even with the extremely simple selection criteria described above, I tended to be authorially satisfied with the generation about 80% of the time. This is perhaps because the generator already takes into account some of the context that an author would consider (and may in part be due to a reader's tendency to pay less attention to satellite sentences than kernel text).

The ideal size of the component grammars was much smaller than was originally anticipated. Most generative grammars have a large enough ruleset to allow for millions of possible expansions. Here, each individual grammar needed only offer a few dozen to a few hundred possible expansions, leading to runtime compound grammars capable of making around five to ten thousand unique sentences. Since each individual grammar in the compound set is run only once or twice during an individual story segment, a smaller number of carefully considered variations proved more effective than a larger space of less curated alternatives.

The system presented is embryonic and addresses only a small part of the problem of creating trustworthy PCG collaborators for interactive story authors. One avenue for future improvement will be finding ways the author can provide more contextual information to the generator without complicating the authoring process. We could shift this burden off the author by integrating the system with existing knowledge bases about the structure of words, sentences, and stories, such as standalone libraries like WordNet or the narration model used in the previously mentioned Curveship. The more context the system has, the less it need rely on randomness when choosing between generation alternatives: WordNet's knowledge of rhymes or syllabic stress, for instance, could weight the selection of satellite sentences with cadence or alliteration matching the author's. Likewise, natural language generation (NLG) techniques could help ensure grammatical and semantic accuracy in generated sentences, a process currently hand-managed by the author through careful construction of expansion grammars.

5. ACKNOWLEDGMENTS

Thanks to Jim Whitehead, who taught the graduate seminar on procedural content generation at UC Santa Cruz which inspired this project, and to Andrew Plotkin for sharing the generation code in "My Secret Hideout."

6. REFERENCES

[1] C. Ashmore and M. Nitsche. The quest in a generated world. In *Proc. 2007 DiGRA: Situated Play*, pages 503–509, 2007.

[2] C. Callaway and J. Lester. Narrative prose generation. *Artificial Intelligence*, 139(2):213–252, 2002.

[3] J. Doran and I. Parberry. Towards procedural quest generation: A structural analysis of RPG quests. *Dept. Comput. Sci. Eng., Univ. North Texas, Tech. Rep. LARC-2010-02*, 2010.

[4] D. F. Harrell. GRIOT's tales of haints and seraphs: A computational narrative generation system. *Second Person*, 2007.

[5] M. Kantrowitz. *Glinda: Natural language text generation in the Oz interactive fiction project.* School of Computer Science, Carnegie Mellon University, 1990.

[6] M. Lebowitz. Creating characters in a story-telling universe. *Poetics*, 13(3):171–194, 1984.

[7] J. Meehan. Tale-spin, an interactive program that writes stories. In *Proc. of the 5th Int'l Conf. on Artificial Intelligence*, 1977.

[8] N. Montfort. Curveship: An interactive fiction system for interactive narrating. *NAACL HLT Workshop on Computational Approaches to Linguistic Creativity*, 2009.

[9] R. Pérez and M. Sharples. Mexica: A computer model of a cognitive account of creative writing. *Journal of Experimental & Theoretical Artificial Intelligence*, 13(2):119–139, 2001.

[10] A. Plotkin. My secret hideout. iOS, 2011. URL: zarfhome.com/hideout/ ; accessed 3-18-12.

[11] A. A. Reed. Perfect. HTML/Javascript, 2011. URL: http://textories.com/perfect/ ; accessed 4-17-12.

[12] C. Seymour. Story and discourse. narrative structure in fiction and film. *Ithaca, NY: Cornell University*, 1978.

[13] A. Sullivan and M. Mateas. Rules of engagement: moving beyond combat-based quests. INT3 '10, pages 11:1–11:8, 2010.