"Generator's Haunted": A Brief, Spooky Account of Hauntological Effects in the Player Experience of Procedural Generation

Max Kreminski
mkreminski@scu.edu
Santa Clara University
Santa Clara, California, USA

ABSTRACT

Theories of the *poetics* of procedural generation attempt to explain the player experience of interacting with generators by describing the aesthetic or experiential qualities that generators can afford when they are deployed in particular ways. We propose that an underinvestigated aspect of procgen poetics—the experiential effects of the sequencing of generated artifacts—can be understood in terms of *hauntology*, a theory of textual interpretation that aims to account for the lingering effects of past texts (and their implied futures) on present ones. We briefly introduce hauntology, discuss a few examples of hauntological effects in player experiences of procgen, and gesture at implications for future technical work.

CCS CONCEPTS

• **Applied computing** → **Computer games**; • **Human-centered computing** → *HCI theory, concepts and models*.

KEYWORDS

procedural generation, player experience, poetics, hauntology

1 INTRODUCTION

Procedural generation is increasingly widely used to create levels, quests, art assets, music, and other "creative facets" of games [10]. Despite this, however, the influence of procgen on player experience (PX)—which is central to the design of games—remains relatively understudied [13], and it is considered difficult both to assess and control PX in the presence of procgen [5].

The theory of *procgen poetics* [6], which aims to describe "the effects that [a] procedural generator can be used to evoke in the player", suggests that it is important for designers to consider not just how individual generated *artifacts* influence PX, but also how PX is changed by the awareness that one is interacting with a

particular kind of generator and by the texture of the generator's *expressive range* [14]. For this reason, procgen poetics is perhaps uniquely well-suited to making sense of PX in games that rely heavily on procgen, including PCG-based games [2].

However, prior discussion of procgen poetics remains "preliminary" [6], with particular gaps around how "interactivity and user involvement"—including the effects of *ordering* as the player encounters a sequence of different artifacts produced by the same generator—influence PX. Closing some of these gaps in procgen poetics would clarify the theory's implications for future technical and design work. Consequently, in this paper, we aim to briefly explain the effects of ordering in the PX of procgen—specifically in terms of *hauntology*.

Hauntology as an approach to textual interpretation was introduced by Derrida [3], originally in response to how Marxism (as an idea that was perceived to be defeated, yet remains both feared and influential) continues to shape critical practice. To oversimplify, hauntology takes the perspective that every *text* (which in modern critical theory can refer to basically any cultural artifact[1]) is necessarily interpreted in light of the "specters" that previous texts conjured into being. Hauntology has been applied to games before, specifically in the context of how players experience narrative in level design [4], but not yet to procgen as far as we are aware.

Here, we first use hauntology to examine a few different ways in which the PX of procgen is dependent not just on the generated artifact (level, quest, weapon, etc) that the player is currently experiencing, but also on the previous generated artifacts that they have encountered in the same play experience and their ever-shifting perception of the generator's expressive range. We frame these phenomena in terms of a bestiary of *specters* by which the player can be haunted during gameplay. Then we briefly discuss the implications of these hauntings for future technical work.

2 A BESTIARY OF SPECTERS

Specters of Individual Artifacts. The simplest form of haunting in the PX of procgen is the form that occurs between artifacts produced by the same generator. In essence, generated artifacts that were encountered later in a play experience are haunted by the ghosts of artifacts that were encountered earlier. A player whose first playthrough of a social simulation game involves (by sheer happenstance) a cast of exclusively *mean* characters, but whose second playthrough involves a more balanced cast, will experience the latter cast very differently than if they had encountered it first. This applies to all kinds of generated artifacts: they are inevitably interpreted in terms of how they are different from, and similar to, those that came before.

---

[1]Generators and generated artifacts among them: see, for instance, Kreminski et al. [8].

Specters of the Expressive Range. Perceived differences and similarities between generated artifacts in turn shape the player's impression of what parameters of variation exist in the generator's expressive range (and what values these parameters can take). The player maintains a *mental model* of the generator and its expressive range, which is shaped by the artifacts they encounter as they encounter them, and this mental model changes its form as the player is exposed to artifacts that the previous mental model could not explain. Artifacts are then interpreted not just in relation to other artifacts, but in relation to an imagined whole space of possible artifacts that tends to grow more aligned with the generator's actual expressive range over time. In other words, the PX is also haunted by the specter of the expressive range. Moreover, the player may act in response to the ghost: for instance, players may pursue the collection of generated inventory items, or build homes in generated locations, that seem rare relative to the overall expressive range.

We see some evidence of this in user studies of procgen-based playful creativity support tools, for instance Nelson et al. [11] and Kreminski et al. [7]: in both studies, some users rapidly gravitated toward creating artifacts that represent extreme points in the generator's expressive range, while others seemed to quickly "map out" the expressive range as a whole before hill-climbing toward a point in that range that most closely matched their creative vision. Both of these patterns of use suggest a strong capacity among users to develop a mostly-accurate mental model of a generator's expressive range when exposed to several of its outputs.

Eerily, if the player's actions shape what artifacts they're exposed to next (for instance, if the player decides to spend time in a rarely-generated enemy-dense region that necessitates them to engage in combat more frequently, leading a behind-the-scenes experience manager to adjust their apparent player type and start generating even more combat encounters), the specter of a particular impression of the expressive range can actually self-reinforce—a recursive sort of haunting wherein a specter is ultimately conjured into being by nothing more than the player's belief in it.

The Specter of Perceptual Collapse. If the generator's expressive range is finite (or if the *perceptual uniqueness* [1, p. 194] of artifacts within its expressive range is finite, which is essentially always the case), the PX is also haunted in some sense by the specter of *perceptual collapse*—the point at which the player has seen a full enough range of perceptually similar artifacts to be confident that "this is all there is". For designers, this is often the point in the PX that we are (implicitly or explicitly) trying to use procgen to forestall, so its apparition is a spooky possibility indeed—and often for players too, the horror of endless repetition (and thus ennui) lurks behind the immediate perception of endless variety.

It is in this sense that the applicability of hauntology to procgen is made most fully manifest: not just players and designers but also *procgen researchers* are haunted by this specter, which we try time and time again (through the development of increasingly sophisticated generative methods) to conjure away. In light of recent developments in generative AI (e.g., text-to-image generators that can deliver seemingly endless streams of distinct outputs), it may seem time to declare this specter banished once and for all. But information-theoretically, the dream of procgen as a means of indefinitely forestalling perceptual collapse without further investment of development effort seems doomed: see, for instance, a recent proof that a generator's expressive range can never be expanded "for free" [12]. Perceptual collapse thus resembles a procgen research equivalent to Derrida's characterization of Marxism [3, p. 48]:

> Today, almost a century and a half later, there are many who, throughout the world, seem just as worried by the specter of communism, just as convinced that what one is dealing with there is only a specter without body, without present reality, without actuality or effectivity, but this time it is supposed to be a past specter. [...] A still worried sigh of relief: let us make sure that in the future it does not come back! At bottom, the specter is the future, it is always to come, it presents itself only as that which could come or come back; in the future, said the powers of old Europe in the last century, it must not incarnate itself, either publicly or in secret. In the future, we hear everywhere today, it must not re-incarnate itself; it must not be allowed to come back since it is past.

Even as we declare victory for our novel generative methods, we yet live in fear of the specter of perceptual collapse: it must not be allowed to come back, because it is past.

3 IMPLICATIONS FOR PROCGEN RESEARCH

This hauntological view of the PX of procgen has two clear implications for future procgen research: one more directly applicable to technical work, one more closely tied to the overarching goals of procgen as a form. First is the implication that the presentation order of generated artifacts represents a key factor in shaping PX. The framework of *experience-driven* procgen [15] attempts to establish connections between procedurally generated artifacts and PX qualities (often via player modeling), but to date the research conducted within this framework has largely treated artifacts as independent of one another. Can we player-model the specter of the expressive range, inferring the player's likely mental model of the generator if they encounter a particular series of generated artifacts? Can we optimize artifact presentation order to prolong the player's perception of endless variety, perhaps by strategically restricting some of the generator's capacity for variety early in a play experience and unlocking it piecemeal so that the expressive range gradually broadens at a rate intended to preserve the player's interest for longer? Both directions may be worth exploring.

The second implication is the spookier of the two, because it may demand rethinking of *what procgen is for*. Even when the player is severely restricted in their capability to request more content from the generator (as in gardening games [9]), mere player awareness that a generator is in operation is often sufficient to summon the specter of perceptual collapse to the scene. Procgen research that embraces rather than fearing this specter may be possible, but how to frame the value of procgen when the dream of infinite free variety is dead and buried remains an open question with which we all might very soon need to contend.

REFERENCES

[1] Katherine Compton. 2019. *Casual Creators: Defining a Genre of Autotelic Creativity Support Systems*. Ph. D. Dissertation. University of California, Santa Cruz.

[2] Michael Cook, Mirjam Eladhari, Andy Nealen, Mike Treanor, Eddy Boxerman, Alex Jaffe, Paul Sottosanti, and Steve Swink. 2016. PCG-based game design patterns. *arXiv preprint arXiv:1610.03138* (2016).

[3] Jacques Derrida. 1994. *Specters of Marx: The State of the Debt, the Work of Mourning and the New International*. Routledge.

[4] Lindsay Grace. 2019. Hauntology, the Penumbra, and the Narratives of Play Experience. In *25th International Symposium of Electronic Art (ISEA), Gwangju, South Korea*.

[5] Christian Guckelsberger, Christoph Salge, Jeremy Gow, and Paul Cairns. 2017. Predicting player experience without the player: An exploratory study. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. 305–315.

[6] Isaac Karth. 2019. Preliminary poetics of procedural generation in games. *Transactions of the Digital Games Research Association* 4, 3 (2019).

[7] Max Kreminski, Isaac Karth, Michael Mateas, and Noah Wardrip-Fruin. 2022. Evaluating Mixed-Initiative Creative Interfaces via Expressive Range Coverage Analysis. In *IUI Workshops*. 34–45.

[8] Max Kreminski, Isaac Karth, and Noah Wardrip-Fruin. 2019. Generators that read. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*.

[9] Max Kreminski and Noah Wardrip-Fruin. 2018. Gardening games: an alternative philosophy of PCG in games. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*.

[10] Antonios Liapis, Georgios N Yannakakis, Mark J Nelson, Mike Preuss, and Rafael Bidarra. 2018. Orchestrating game generation. *IEEE Transactions on Games* 11, 1 (2018), 48–68.

[11] Mark J Nelson, Swen E Gaudl, Simon Colton, and Sebastian Deterding. 2018. Curious users of casual creators. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*.

[12] Younès Rabii and Michael Cook. 2023. Why Oatmeal is Cheap: Kolmogorov Complexity and Procedural Generation. In *Proceedings of the 18th International Conference on the Foundations of Digital Games*.

[13] Gillian Smith. 2014. Understanding procedural content generation: a design-centric analysis of the role of PCG in games. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 917–926.

[14] Gillian Smith and Jim Whitehead. 2010. Analyzing the expressive range of a level generator. In *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*.

[15] Georgios N Yannakakis and Julian Togelius. 2011. Experience-driven procedural content generation. *IEEE Transactions on Affective Computing* 2, 3 (2011), 147–161.