Gardening Games: An Alternative Philosophy of PCG in Games

Max Kreminski
University of California Santa Cruz
Department of Computational Media
Santa Cruz, CA
mkremins@ucsc.edu

Noah Wardrip-Fruin
University of California Santa Cruz
Department of Computational Media
Santa Cruz, CA
nwardrip@ucsc.edu

ABSTRACT

Procedural content generation (PCG) in games is often framed as a way to *feed the content furnace*, satisfying the voracious appetites of players by generating infinite seas of content for them to consume. Although this dominant framing provides a clear structuring purpose for PCG research, it also unnecessarily limits our vision of alternative purposes that generative methods might serve. Furthermore, generative systems designed with this purpose in mind may tend to reinforce certain problematic dynamics in game design. In this paper, we draw a contrast between two approaches to procedural terrain generation and the dynamics of play they tend to enable, which we term *mining* and *gardening*. We then extend this analysis to PCG more broadly and suggest that the latter (gardening) dynamic represents a viable and compelling alternative philosophy of how generative methods can be used in games.

CCS CONCEPTS

• **Software and its engineering → Interactive games**

KEYWORDS

procedural content generation, generative methods

ACM Reference format:

Max Kreminski and Noah Wardrip-Fruin. 2018. Gardening Games: An Alternative Philosophy of PCG in Games. In *Proceedings of FDG'18, Malmö, Sweden, August 2018,* 2 pages.

1 INTRODUCTION

What is the purpose of procedural content generation (PCG) in games? As Compton et al. [1] have pointed out, PCG is most commonly framed as a kind of cost-saving measure for game designers: in order to satisfy the voracious appetites of players for more content, designers delegate some of the work of creating new content to a generative system, which can churn out a hypothetically infinite amount of content to keep players satisfied.

This approach to PCG in games has seen significant commercial success in games such as *Minecraft*, which uses procedural terrain generation to keep players indefinitely occupied with the exploration and exploitation of a potentially infinite world. At the same time, however, this approach is not without its problems. If the artifacts produced by a game's generator are insufficiently *perceptually unique*, the game may not hold player interest for long, regardless of the sheer number of mathematically distinct artifacts it can generate [2]. Moreover, *Minecraft* and similar games have been critiqued as "entitlement simulators" [4] that uncritically replicate the logic of colonialism [3].

We describe *Minecraft* as a *mining game*: a game in which the fundamental dynamic of gameplay is *extractive* and PCG is employed primarily to generate an *infinite frontier*, ensuring that there will always remain fresh content for the player to explore and exploit. Mining games are characterized by their treatment of generated artifacts as readily disposable, and – in particular – by their willingness to generate new content (such as terrain in *Minecraft*) whenever the player requests it.

Mining games stand in direct opposition to *gardening games*: an alternative variety of games that also often make use of generative methods as a fundamental element of their design. Unlike mining games, however, gardening games (exemplified by *Animal Crossing*) are characterized by the way in which they treat generated artifacts as *non*-disposable. Gardening games encourage players to engage deeply with a relatively small number of generated artifacts, and make use of generative methods to gradually adapt or develop these few "seed" artifacts over time.

In this paper, we will briefly compare procedural terrain generation in *Minecraft* and *Animal Crossing* to characterize the key features of mining and gardening games respectively. We will then explore how gardening games represent one possible solution to the issues of perceptual uniqueness and colonialist dynamics in heavily PCG-reliant game design.

2 CASE STUDY: MINING VS GARDENING

2.1 *Minecraft* as Mining Game

In *Minecraft*'s survival mode, much of gameplay revolves around the extraction of resources from generated terrain. As the player spends more time in a particular area of the game world, their extraction of resources and exploration of available "content" in the area (which mostly takes the form of recognizable generated structures, such as dungeons) tends to make the area feel progressively less interesting and less alive. Then, once the player has mined out one area to the extent that it no longer interests them, they move on to the next.

Even if the player chooses to establish a "home base" to which they may repeatedly return, improvement of the base often makes use of resources mined out from the rest of the world. As non-renewable resources become scarcer within the base's immediate vicinity, players end up venturing further and further afield to continue obtaining the resources they need to expand.

In order to extend this dynamic indefinitely, enabling players to continue moving from area to area without ever having to face

any consequences of this extractive logic, *Minecraft* employs PCG to generate an *infinite frontier,* ensuring that there will always remain new spaces for the player to explore and exploit. New terrain is generated whenever the player approaches the edge of existing terrain, enabling the player to continue journeying and extracting resources indefinitely in any direction as they please.

The line between gardening and mining may not always be completely clear, and some games may feature elements that favor both mining and gardening dynamics. *Minecraft*, for instance, may be played non-extractively. Although the typical mode of play treats individual "chunks" of generated terrain as disposable, some players nevertheless prefer to play differently; they may, for instance, settle within an NPC "village" and try to protect rather than kill its inhabitants, or elect to make use of only renewable materials for construction. However, the game mechanically favors one dynamic over the other, using PCG to enable and even encourage an extractive mode of play.

2.2 *Animal Crossing* as Gardening Game

Like *Minecraft*, *Animal Crossing* makes use of procedural terrain generation. Unlike *Minecraft*, however, *Animal Crossing* runs its terrain generator exactly once: the first time the player launches the game. Once established, the play space remains tightly bounded; the player may not travel beyond the bounds of the world as it was initially generated.

How, then, does *Animal Crossing* sustain player interest? Rather than employing procedural terrain generation to distribute interesting content across space, *Animal Crossing* makes use of generative methods driven by the real-world clock and calendar to gradually develop certain aspects of the town, thereby distributing interesting player-visible variation in content over *time*. Through repeated interaction with the town, the player becomes deeply familiar with a single generated artifact; their story of play becomes the story of sustained engagement with a living and growing place over time.

Aside from the initial terrain generation step, the generative methods that *Animal Crossing* employs may not be readily recognizable as "generators" in the traditional sense. Rather than one large and highly visible centralized "generator", *Animal Crossing* employs a variety of tiny generative processes that act largely independently of one another to perform simple tasks like placing weeds and buried treasure, populating the town with the appropriate fish and bugs for a given set of environmental conditions, determining which animal residents should move in or out of the village, and so on. Furthermore, rather than generating a single clearly defined artifact in a single discrete step, these generative processes are instead used to gradually adapt the existing "seed" artifact that is the town.

3 WHY GARDENING GAMES?

3.1 Perceptual Uniqueness

One recurring issue in PCG research involves the problem of perceptual uniqueness. The need to generate perceptually unique artifacts is severely exacerbated by games that must present players with wholly new content as rapidly as the player is able to request it: games, in other words, where mining dynamics prevail. A game in which players are free to pass by or dispose of generated content whenever they see fit, with the expectation that more content will immediately be generated to fill its place, puts a severe burden on its generators to continue producing perceptually unique artifacts at the speed of player consumption.

Gardening dynamics represent a design-based solution to this problem. If you were to generate hundreds of *Animal Crossing* towns, place them side by side, and consider them in aggregate, you would be unlikely to judge any one of them as particularly unique. Yet it is common for players of *Animal Crossing* to develop a strong sense that *their* town is somehow special. This sense of uniqueness develops as a result of sustained, deep engagement with a single, gradually evolving generated artifact, with the player's decisions having a degree of impact on the artifact's development. This kind of sustained, deep engagement is all but impossible in games where generated content is readily consumed or disposed of shortly after it is first generated.

3.2 Colonialism & the "Entitlement Simulator"

Mining games characteristically treat generated artifacts as disposable, enabling and arguably encouraging dynamics of extraction and consumption. Furthermore, because generation of fresh content in mining games occurs automatically whenever the player exhausts the content already available to them, the world is placed in a fundamentally subservient position to the player: it exists solely to serve the player's needs.

Gardening games replace these dynamics of extraction and consumption with dynamics of nurturing and caretaking. They also tie generation of fresh content to forces totally out of the player's control (such as the progression of the real-world calendar), thereby somewhat de-centering the needs of the player relative to those of the game world and placing them on closer to a level footing with one another.

4 CONCLUSIONS

Gardening games represent a potential alternative philosophy of how PCG can be used in games, presenting solutions to several recurring problems in heavily PCG-reliant game design. To benefit from this philosophy, however, we must be willing to adopt an expansive definition of generative methods: one that recognizes generative processes that gradually adapt existing artifacts rather than creating new ones as objects worthy of study.

REFERENCES

[1] Kate Compton, Joseph C. Osborn, and Michael Mateas. 2013. Generative methods. In *The Fourth Procedural Content Generation in Games workshop, PCG.*
[2] Kate Compton. 2016. So you want to build a generator. Retrieved from http://galaxykate0.tumblr.com/post/139774965871/so-you-want-to-build-a-generator.
[3] Daniel Dooghan. 2016. Digital conquerors: Minecraft and the apologetics of neoliberalism. *Games and Culture* (29 June 2016). DOI: https://doi.org/10.1177%2F1555412016655678
[4] Meg Jayanth. 2016. Forget protagonists: writing NPCs with agency for 80 Days and beyond. Retrieved from https://medium.com/@betterthemask/forget-protagonists-writing-npcs-with-agency-for-80-days-and-beyond-703201a2309.