Art and Science of Engineered Design: What Kind of Discipline is PCG?

Jim Whitehead
Dept. of Computational Media, UC Santa Cruz
University of California, Santa Cruz
Santa Cruz, CA 95064
ejw@soe.ucsc.edu

ABSTRACT

What kind of discipline is PCG? PCG research can be viewed as science, engineering, design, and art. PCG is thus a multidiscipline, drawing from a broad set of epistemic traditions.

CCS CONCEPTS

• **Computing methodologies → Philosophical/theoretical foundations of artificial intelligence**;

KEYWORDS

Philosophy of procedural content generation, philosophy of science

ACM Reference format:
Jim Whitehead. 2017. Art and Science of Engineered Design: What Kind of Discipline is PCG?. In *Proceedings of FDG'17, Hyannis, MA, USA, August 14-17, 2017,* 3 pages.
https://doi.org/10.1145/3102071.3110571

1 INTRODUCTION

Procedural content generation (PCG) is a field of study focused on "the algorithmic creation of game content with limited or indirect user input" where game content consists of "levels, maps, game rules, textures, stories, items, quests, music, weapons, vehicles, characters, etc." [9]. It is an unusual discipline. Whereas scientific fields tend to study *naturally* occurring objects, systems, and their behaviors, in PCG the goal is to create *synthetic* objects and systems. Whereas engineering is concerned with the *human* application of scientific knowledge to create artifacts to solve human problems, in PCG humans create generators so that *computers* can make things. Design fields focus on the creation of artifacts, but by humans, often with a greater emphasis on their aesthetics and values than engineering. In PCG, aesthetics and values have traditionally been secondary concerns. Finally, while generative art and PCG share many commonalities, the goal of generative art is to create artworks, and not to create purposeful pieces of a game experience.

FDG'17, August 14-17, 2017, Hyannis, MA, USA

ACM ISBN 978-1-4503-5319-9/17/08…$15.00
https://doi.org/10.1145/3102071.3110571

PCG does not entirely belong to any of the fields of art, science, design or engineering. But neither is it entirely divorced from them. PCG draws on methods and ways of thinking from all of these disciplines, and projects emphasize them in differing amounts. This begs the question: what kind of discipline is PCG?

2 THE SCIENCE OF PCG

PCG is not a science in the traditional sense, as there is no natural phenomena that that the field focuses upon and seeks to explain. The role of science in generative methods more broadly can be explored by contrasting the aims of the Computational Creativity community with that of the PCG community. Within Computational Creativity, a community goal is to develop generative systems that embody models of human or artificial creativity [4]. The phenomena of interest is creativity, and generative systems are useful and valuable to the extent they explore novel models of creative construction. Hence, Computational Creativity is engaged in scientific inquiry into how creativity occurs, with experiments performed using executable models of creativity. In contrast, PCG research typically makes no claim to being a model of creativity: the focus is on the qualities of the generated artifact. A PCG system has value to the extent it can generate artifacts of high quality, with applicability in industrial contexts being a laudable goal. There is a utilitarian bent to the PCG community.

Herbert Simon in *The Sciences of the Artificial* argues that our world is increasingly comprised of human created artificial objects, not just natural ones. The means by which these artificial objects can be optimally created is amenable to scientific analysis. Hence, a science of the artificial is a *science of design*[7]. Within PCG, the most pure form of this science of design is development of novel generation algorithms. This work usually takes the form of adapting some existing technique for the context of PCG. Examples include the marriage of Chomsky-style grammars with turtle graphics to create L-systems, or the first application of genetic algorithms to content generation. This is pure discovery, a dive into the ocean of ideas and algorithms to bring back useful techniques for synthesis.

PCG also engages in pure design science when it explores novel theories of design via the construction of executable models of the design activity, in much the same way computational creativity systems create executable models of creativity. The Launchpad system for platformer level generation used this approach with its executable model of level design centered on player action rhythm [8]. Less obviously scientific is the common situation where PCG research takes an existing generation technique and applies it in an innovative way. Launchpad exemplifies this too, with the rhythm-based design model being realized via grammar productions.

This is where things get blurry. Engineering and design both involve the application of known design techniques for the construction of artifacts that meet specific goals and human needs. At what point is the use of an existing generative method sufficiently innovative that it is considered science, as opposed to use of a generative method that is skillful application of one of the many tools in the PCG toolbox, and hence engineering or design?

3 PCG AS ENGINEERING

Engineering is generally viewed as the application of scientific knowledge for the construction of artifacts that satisfy multiple goals. Research in engineering is interested in issues such as more optimal use of design and implementation techniques, techniques for creating more optimal artifacts, approaches for validating that created artifacts meet desired goals, and methodologies (engineering processes) for the creation of artifacts. PCG research engages most of these issues.

Skillful use of existing generative methods for the construction of artifacts is a core research topic in PCG. Such work deeply analyses some form of artifact, then deconstructs it in a way that is amenable for a generative method. An example of this is the use of genetic algorithms for the construction of dungeons [1]. The research involved skillful representation of dungeons and creation of functions for evaluating them using genetic algorithms, but did not involve the creation of fundamentally new generative methods. It is a useful, pragmatic result: if one wants to create dungeons, this approach will work.

Some PCG research explores how to validate created artifacts. This line of research focuses on developing appropriate metrics for evaluating generated content, both manually and automatically. This can be viewed as similar to engineering subfields concerned with measuring and evaluating engineered artifacts. Within PCG, the work on platformer level metrics [6, 8] and static and dynamic analysis of platformer levels [2] exemplifies this kind of engineering metrics research.

Viewing PCG as an engineering activity has strong explanatory power: much PCG research can be viewed as creating engineering knowledge. But, just as clearly, reducing all PCG work to be forms of engineering is too simple. The game *No Man's Sky* (Hello Games, 2016) is an interesting example. On one hand, its content generation methods can be viewed as engineering, scaling up known techniques to meet the size and performance goals of a AAA game. But, this engineering frame sidelines the inherent beauty of these generated worlds: it is equally valid to view the game as generative art. When one of the goals to be satisfied by an artifact is aesthetic beauty, we're suddenly out of the realm of engineering.

4 PCG AS DESIGN

There are multiple meanings of design active within PCG. One is the notion coming from design research, that of incorporating human needs and aesthetics into the process of constructing an artifact. While similar to engineering in its emphasis on constructing artifacts that meet multiple goals, design research is more open to end-user engagement in the design process, and the importance of beauty in designed objects. Mixed-initiative content creation systems can be viewed as a form of PCG as design research, with their focus on enabling design processes comprised of cooperating human and computer designers [5].

An alternate notion of design is that of game design, constructing the many interlocking systems that comprise a working game. *Galactic Arms Race* is an example of PCG as game design [3]. This game evolves novel particle systems which serve as the weapons for player spaceships in a 2D space shooter. By one account, this game can be viewed as an engineering activity: the application of known techniques for evolving neural network-based particle systems. However, this hides the strongest contributions, that of building utility evaluation and selection into the mechanics of the game. This game is best viewed as innovative game design research. But, as with *No Man's Sky*, it would be overly simple to view *Galactic Arms Race* only through the lens of game design research: its generated particle systems are also beautiful, and hence can be viewed as an artistic contribution too.

5 PCG AS ART

In the best case, such as *No Man's Sky* and *Galactic Arms Race*, procedurally generated content could be extracted from its game context and displayed as a compelling artwork. The need for generated content to be visually compelling to meet AAA game expectations means that PCG work is often also generative art research. As an artwork, it is judged on its visual aesthetic qualities, and on any artist statements produced by the development team. There is also an aesthetics of the overall ensemble of created artifacts, subject to critique. Procedurally generated planets in *No Man's Sky* embody an aesthetic of the infinite sublime (positive view) or suffer from the sameness of a thousand bowls of oatmeal (negative view). Both are clearly artistic judgements.

6 PCG AS MULTIDISCIPLINE

So, again, what kind of discipline is PCG? Having wrestled with this question, we can now provide preliminary answers. Perhaps the best way of viewing PCG is as a multidiscipline. Across the field we see research that is characterized as science, engineering, design, art, or their combinations. This diversity of intellectual traditions is a source of strength for PCG, giving it rich intellectual vibrancy. They mark PCG as distinctive as well, since few fields draw so deeply from such breadth. Only architecture and design share the same emphasis on creating artifacts that are both functional and beautiful, yet PCG's focus on (semi-)automatic generation sets it apart from these two fields. This is the unique mission of PCG: a field engaging a broad set of epistemic traditions while focused on automatically creating content useful for games.

REFERENCES

[1] Daniel Ashlock, Colin Lee, and Cameron McGuinness. 2011. Search-Based Procedural Generation of Maze-Like Levels. *IEEE Trans. Computational Intelligence and AI in Games (TCIAIG)* 3, 3 (Sept. 2011).
[2] Matthew Guzdial, Nathan Sturtevant, and Boyang Li. 2016. Deep Static and Dynamic Level Analysis: A Study on Infinite Mario. In *Proc. 2016 Workshop on Experimental AI in Games (EXAG'16)*.
[3] Erin Hastings, Ratan K. Guha, and Kenneth O. Stanley. 2009. Automatic content generation in the galactic arms race video game. *IEEE Trans. Computational Intelligence and AI in Games (TCIAIG)* 1, 4 (Dec. 2009).
[4] Anna Jordanous. 2011. Evaluating evaluation: Assessing progress in computational creativity research. In *Proc. 2nd Int'l Conf. on Comp. Creativity (ICCC-11)*.
[5] Antonios Liapis, Gillian Smith, and Noor Shaker. 2016. Mixed-initiative content creation. In *Procedural Content Generation in Games*. Springer.

[6] Julian R. H. Mariño, Willian M. P. Reis, and Levi H. S. Lelis. 2015. An Empirical Evaluation of Evaluation Metrics of Procedurally Generated Mario Levels. In *Proc. AIIDE 2015*. 44–50.
[7] Herbert Simon. 1996. *The Sciences of the Artificial* (3rd ed.). MIT Press.
[8] Gillian Smith, Jim Whitehead, Michael Mateas, Mike Treanor, Jameka March, and Mee Cha. 2011. Launchpad: A Rhythm-Based Level Generator for 2-D Platformers. *IEEE Trans. Comp. Intelligence and AI in Games (TCIAIG)* 3, 1 (March 2011).
[9] Julian Togelius, Noor Shaker, and Mark J. Nelson. 2016. Introduction. In *Procedural Content Generation in Games*. Springer.