The Randomizer Community does Procedural Content Generation Research

Ross Mawhorter
rmawhort@ucsc.edu
University of California Santa Cruz
Santa Cruz, CA, USA

Peter Mawhorter
pmawhort@wellesley.edu
Wellesley College
Wellesley, Massachusetts, USA

Adam Smith
amsmith@ucsc.edu
University of California Santa Cruz
Santa Cruz, CA, USA

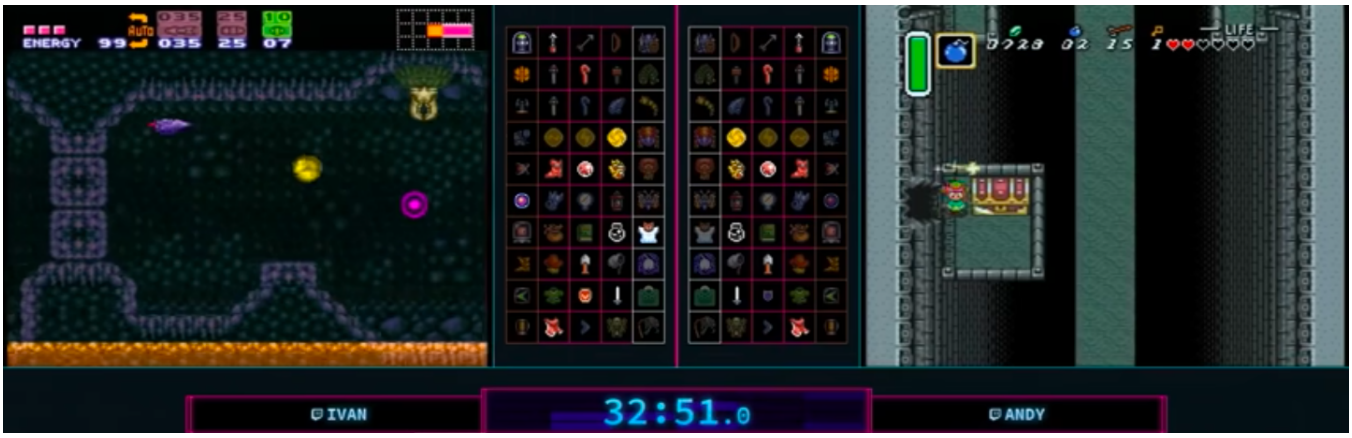

Figure 1: Screenshot from the Super Metroid/A Link to the Past combo randomizer run at Summer Games Done Quick 2019 [6].

ABSTRACT

Academic Procedural Content Generation research has until recently overlooked a significant real-world application of generative methods to existing games: game randomizers. These programs remix existing games by changing things like item locations, enemy stats, or even room connections to create a fresh experience based on a beloved game, and are especially popular among speedrunning and streaming communities. They generate where high-production-quality full-scale games, explicitly geared towards replay value. Randomizers fulfill many of the stated motivations of the academic PCG research community, and important new research directions can be developed by investigating this space.

CCS CONCEPTS

• **Applied computing** → **Media arts**; • **Theory of computation** → *Generating random combinatorial structures*; • **Human-centered computing** → HCI theory, concepts and models.

KEYWORDS

procedural content generation, randomizers, playability

PCG Workshop '22, September 5, 2022, Athens, Greece

ACM ISBN 978-1-4503-XXXX-X/18/06…$0.00
https://doi.org/XXXXXXX.XXXXXXX

ACM Reference Format:
Ross Mawhorter, Peter Mawhorter, and Adam Smith. 2022. The Randomizer Community does Procedural Content Generation Research. In *Proceedings of Procedural Content Generation Workshop (PCG Workshop '22).* ACM, New York, NY, USA, 3 pages. https://doi.org/XXXXXXX.XXXXXXX

1 WHY RANDOMIZERS?

Figure 1 shows two players cooperatively playing the *A Link to the Past and Super Metroid Combo Randomizer* [12]. This is an automatically generated mod for these two games where players can travel between the games, and items from both games have been moved, with logic to ensure that the games can be completed. This is one example of a game "randomizer", a program that remixes the content of an existing game, while ensuring that the game remains playable. These modding programs are actively in use and popular among speedrunning and streaming communities, especially for Metroidvania and RPG games where randomizing just a few aspects of the game can create a fresh experience that tests the skills of veteran players. Many randomizers are reliable enough that newly-generated games are used in tournament races with hundreds of spectators [16]. Players can trust that the algorithm's output is always playable, even if the generated games often take over an hour of continuous gameplay to complete. Yet the PCG research community has largely overlooked the randomizer community, despite their ability to achieve such oft-cited aims. This paper argues that we should cite randomizer developers and deeply examine their work in order to understand how their work makes progress on some longstanding questions in PCG research. This is an opportunity for collaboration and crosstalk in the same vein as the Roguelike celebration, which has both helped disseminate

academic research into the wider games community and inspired further research based on the needs of roguelike developers.

2 WHAT ARE RANDOMIZERS?

To discuss randomizers in more detail, we will focus on the *A Link to the Past Item Randomizer*, or ALTTPR [15]. As input, ALTTPR takes an original game ROM for the Japanese version of *A Link to the Past* [13]. In addition, the user specifies settings such as what glitches the player should be expected to know in order to reach item locations and whether or not to place dungeon items outside of the dungeon they are required for. The output is a randomized game ROM and a spoiler file, which lists the changes made to the ROM, as well as an example order of item locations to visit in order to beat the game. Players either use a SNES emulator or original SNES hardware to play the randomized game. The ROM is stamped with the RNG seed on the menu, so that players who are racing each other can verify that they are playing the same randomized game.

In order to generate a completable game, the randomizer begins by placing a "progression item" at a location the player can reach from the start of the games with no items. This progression item unlocks further item checks, and additional progression items are placed until the final boss can be reached. In existing terms, this can be viewed as simultaneously generating the mission and the space, while considering how the player can backtrack to use newly gained abilities in earlier areas [2]. In addition to merely changing the locations of in-game items, ALTTPR and other randomizers also have the ability to create an *entrance randomizer*, where doors within the game lead to different locations (while still ensuring playability). Other prominent examples include randomizers for *Super Metroid* [3, 11] and *Hollow Knight* [1, 5].

3 RESEARCH WITH RANDOMIZERS

There are many research directions that involve PCG research and randomizers.

Understanding Randomizers: There are some critical questions about how randomizers fit into the broader picture of games, and how they relate to existing PCG methods. Some existing work examines the player communities of randomizers [8], as well as an in-depth analysis of a single randomizer as a game itself [7]. However, we are not aware of any existing player experience research focused on randomizers. Clear directions for future research include understanding the motivations behind randomizer play (which might help explain their popularity), comparing player experiences between randomized and original games, and looking at how gameplay strategies differ from non-randomized games given the large difference in information availability (players typically know the entire map layout ahead of time, just not which items are where).

Creating Randomizers: Building new randomizers is another important research direction. Randomizers are different than many academic settings for PCG. Generated levels must be guaranteed to be completable, and are intended to be played for hours. This contrasts with the scale of existing systems, and requires rethinking some of the basic assumptions about what types of content our systems should generate. Another critical difference is that many randomizers are played in a tournament setting as a race. New generators may explicitly target that setting, and create exciting situations intentionally. PCG research can also aid directly in randomizer development. Most randomizers are built on a hand-written logic that details how the player can move through the game. Extracting this type of logic automatically might be possible with insights from existing automated playtesting systems.

Learning From Randomizers: PCG researchers can also learn game-specific domain knowledge from randomizer developers in order to apply more traditional techniques to these existing popular games. This saves researchers from having to develop high-quality games from scratch, while also ensuring that the generated content has a wider public audience. This idea has been applied as a proof-of-concept for *Super Metroid*, but can also extend to other games [9]. Targeting existing games for PCG output also serves to improve the ecological validity of procedural level design research. Rather than being able to design out certain complicated cases, researchers must grapple with the complexity of a real commercial game. This in turn helps ensure that the resulting methods will be easier to cross-apply.

Using Existing Randomizers: Randomizers also provide opportunities for other research that has used PCG outputs. Since randomizer races are livestreamed, there is a large source of human gameplay data, with multiple playthroughs of the same underlying game. Randomizers are seeded, and the race seeds are known, meaning that the same randomized game can be re-generated after the fact. This creates a unique opportunity for learning from human data (which is already a promising research area [4, 10]). Randomizers are also a good potential playground for training and testing new AI systems (where domain randomization is already studied, see e.g., [14]). Because the item locations are different each time, agents that can learn to play randomizers cannot simply memorize a solution.

4 CONCLUSION

Increased attention to randomizers as PCG research would benefit the academic community because:

- Research with randomizers has good ecological validity
- Randomizer-based projects can be more ambitious given limited development time
- Randomizers deploy understudied methods to achieve strong playability guarantees
- Randomizers have active player audiences who would be amenable to participating in PCG experiments

The people who have developed existing randomizers are indeed PCG researchers even though they may not have any ties to academia, and more dialogue with them could be fruitful for both communities.

REFERENCES

[1] Team Cherry. 2017. Hollow Knight. Team Cherry. PC.
[2] Joris Dormans. 2010. Adventures in Level Design: Generating Missions and Spaces for Action Adventure Games. In *Proceedings of the 2010 Workshop on Procedural Content Generation in Games* (Monterey, California) *(PCGames '10)*. Association for Computing Machinery, New York, NY, USA, Article 1, 8 pages. https://doi.org/10.1145/1814256.1814257
[3] flogh and dhmdmdhm et al. 2018. Super Metroid VARIA Randomizer. https://randommetroidsolver.pythonanywhere.com/

[4] Christoffer Holmgård, Michael Cerny Green, Antonios Liapis, and Julian Togelius. 2019. Automated Playtesting With Procedural Personas Through MCTS With Evolved Heuristics. *IEEE Transactions on Games* 11, 4 (2019), 352–362. https://doi.org/10.1109/TG.2018.2808198
[5] homothetyhk and seresharp et al. 2018. Hollow Knight Randomizer 3. https://github.com/homothetyhk/HollowKnight.RandomizerMod
[6] Samuel Horti. 2019. Summer Games Done Quick 2019 smashes GDQ fundraising record. https://www.pcgamer.com/summer-games-done-quick-2019-smashes-gdq-fundraising-record/
[7] Michael Iantorno. 2021. See You Next Mission. *ROMchip* 3, 2 (2021).
[8] Celeste Clark Jewett. 2021. *Game Randomization: Emergent Gameplay Communities.* Ph. D. Dissertation. University of California, Santa Cruz.
[9] Ross Mawhorter, Batu Aytemiz, Isaac Karth, and Adam Smith. 2021. Content Reinjection for Super Metroid. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 17. 172–178.
[10] Johannes Pfau, Antonios Liapis, Georgios N. Yannakakis, and Rainer Malaka. 2022. Dungeons & Replicants II: Automated Game Balancing Across Multiple Difficulty Dimensions via Deep Player Behavior Modeling. *IEEE Transactions on Games* Early Access (2022), 1–1. https://doi.org/10.1109/TG.2022.3167728
[11] Yoshio Sakamoto. 1994. Super Metroid. Nintendo. Super Nintendo Entertainment System.
[12] tewtal and RebelusQuo et al. 2019. Super Metroid and A Link to the Past Crossover Randomizer. https://samus.link/
[13] Takashi Tezuka. 1991. The Legend of Zelda: A Link to the Past. Nintendo. Super Nintendo Entertainment System.
[14] Joshua Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and P. Abbeel. 2017. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 23–30.
[15] Veetorp, KatDevsGames, ChristosOwen, and Smallhacker et al. 2017. The A Link to the Past Randomizer. https://alttpr.com/en
[16] Scott White. 2019. How "randomizers" are breathing new life into old games. https://arstechnica.com/gaming/2019/12/how-randomizers-are-breathing-new-life-into-old-games/