

General Statistical Approaches to Procedural Map Generation

Sam Snodgrass

Drexel University, Department of Computer Science
Philadelphia, PA, USA
Sps74@Drexel.edu

Abstract

Procedural content generation (PCG) studies the algorithmic creation of content (e.g., textures, maps), typically for games. PCG has become a popular research topic in recent years, but little has been done in terms of generalized content generators: approaches that can generate content for a variety of games without hand-tuning. We are interested in exploring statistical algorithms that could lead to generalized procedural map generators.

1 Introduction

Generating video game content algorithmically (procedural content generation or PCG), allows players to experience new and unique content. Most PCG research focuses on domain-specific approaches. We are trying to address the problem of generalizable approaches, or approaches that could be applied to multiple domains without hand-tuning. In particular, we are interested in statistical approaches. There recently has been increased interest in using statistical approaches to model and produce video game maps [Dahlskog *et al.*, 2014; Guzdial and Riedl, 2015]. Our goal is develop these statistical approaches to be used across many domains.

1.1 Procedural Map Generation

Procedural content generation (PCG) studies the algorithmic creation of content [Shaker *et al.*, 2015], typically for video games. This section discusses techniques for map generation in 2-D platforming games (e.g., *Super Mario Bros.*).

We are interested in generators that learn statistical properties from training maps, and use them to sample new maps. Some interesting work in this area includes sampling new maps using n -grams trained on input maps [Dahlskog *et al.*, 2014], as well as generating maps using a statistical model trained on gameplay footage [Guzdial and Riedl, 2015].

In addition to the techniques above, there are many other techniques for generating 2-D platforming game maps. For example, Smith *et al.* experimented with a rhythm-based approach [Smith *et al.*, 2009], and Mawhorter and Mateas explored occupancy-regulated extension, a geometry assembly approach [Mawhorter and Mateas, 2010]. There have even been map generation competitions for 2-D platforming games [Shaker *et al.*, 2011]. However, most of the above approaches

are domain-specific, because designing domain-independent PCG approaches is an open problem.

2 Research Plan

We plan to address the problem of domain-independent PCG approaches by developing statistical algorithms to learn models of maps from training data, and create new maps using those learned models. Our plan includes three steps:

1. **Representation Formalism:** Maps from different games often have different representations. For example, *Spelunky* and *Diablo III* are dungeon exploration games, but *Spelunky* uses a 2-D, retro-style, side view map with squares as the building blocks, whereas *Diablo III* uses a 3-D, semi-realistic, top-down view map. Can we define map representation formalisms which produce uniform map representations, and can capture the important aspects of maps for a wide variety of games?
2. **Learning Algorithms:** We need to devise new statistical algorithms for training models from data, and sampling new maps. What types of models are applicable to map generation, and how can we apply them to our new map representations?
3. **Evaluation:** Metrics must be developed to measure the quality of the generated maps and the performance of the algorithms. Which domains should be considered for testing our algorithms, and what attributes of the maps and algorithms can be used to measure quality?

3 Progress

3.1 Representation Formalism

We represent input maps using tile arrays. Given a map, we identify the different objects within the maps (e.g., ground, enemies, power-up blocks), and re-represent the input maps using those tile types, yielding a multi-dimensional array of tile types. Note, for the chosen games the arrays are 2D, but this can be extended to 3D arrays for 3D games. Figure 1 shows a section of a *Super Mario Bros.* map and how it is represented with tiles. Additionally, we have explored the use of hierarchical map representations, where maps are represented at multiple levels. The low-level is as before, but an additional high-level representation is created, where each high-level tile corresponds to some structure or pattern created us-

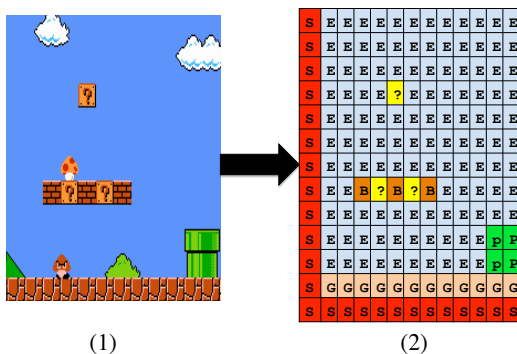


Figure 1: Representation of a *Super Mario Bros.* map section.

ing the low-level tiles [Snodgrass and Ontanon, 2014]. Exploring multi-layered hierarchical representations and higher-dimensional game maps is part of our future work.

3.2 Learning Algorithms

We are exploring Markov models for learning a model from a set of maps. We use a multi-dimensional extension of Markov chains (MdMCs) whose network structure includes several previous states, where previous states can occur in a multi-dimensional grid. We also experimented with Markov random fields (MRFs). We found that MdMCs and MRFs are able to model two-dimensional training data, such as maps.

We have developed new sampling algorithms for MdMCs, which are suitable for PCG. One algorithm probabilistically places the next tile in a map based on a trained model and the surrounding tiles. The algorithm performs a lookahead function that is used to avoid configurations of tiles that have not been encountered during training. If such a state is unavoidable, the algorithm falls back to a simpler model and attempts resampling the tile. Another algorithm we are exploring, uses constraints during sampling in order to guide the output.

3.3 Evaluation

We are using *Super Mario Bros.*, *Loderunner*, and *Kid Icarus* as the initial testing domains for the above algorithms. *Super Mario Bros.* and *Kid Icarus* are both platforming games, and *Loderunner* is a puzzle-platforming game, where paths are very important within the maps. In the future we plan on testing with more complex games, such as *Metroid*.

We use several metrics to evaluate the generated maps and the chosen algorithm. For the maps, we test the playability and well-formedness, as well as the linearity and leniency. For the algorithms, we explore the range of the variety of maps that the algorithms produce, and the time taken to create new maps. Additionally, for this conference we explored the use of domain-specific constraints in order to produce maps of a desired type. We are investigating more evaluation metrics for both the maps and the algorithms, as well as performing user studies in order to gather human feedback.

4 Conclusion

Statistical procedural content generation approaches have been gaining attention recently, but little has been done to-

wards generalized content generators. Statistical models are a viable option for generalized map generators, and we have made progress by employing Markov models and an abstract map representation.

References

- [Dahlskog *et al.*, 2014] Steve Dahlskog, Julian Togelius, and Mark J Nelson. Linear levels through n-grams. *Proceedings of the 18th International Academic MindTrek*, 2014.
- [Guzdial and Riedl, 2015] Mathew Guzdial and Mark Riedl. Toward game level generation from gameplay videos. In *FDG 2015*, 2015.
- [Mawhorter and Mateas, 2010] Peter Mawhorter and Michael Mateas. Procedural level generation using occupancy-regulated extension. In *Computational Intelligence and Games (CIG), 2010 IEEE Symposium on*, pages 351–358. IEEE, 2010.
- [Shaker *et al.*, 2011] Noor Shaker, Julian Togelius, Georgios N Yannakakis, Ben Weber, Tomoyuki Shimizu, Tomonori Hashiyama, Nathan Sorenson, Philippe Pasquier, Peter Mawhorter, Glen Takahashi, et al. The 2010 mario AI championship: Level generation track. *TCIAIG, IEEE Transactions on*, 3(4):332–347, 2011.
- [Shaker *et al.*, 2015] Noor Shaker, Julian Togelius, and Mark J. Nelson. *Procedural Content Generation in Games: A Textbook and an Overview of Current Research*. Springer, 2015.
- [Smith *et al.*, 2009] Gillian Smith, Mike Treanor, Jim Whitehead, and Michael Mateas. Rhythm-based level generation for 2D platformers. In *Proceedings of the 4th International Conference on Foundations of Digital Games*, pages 175–182. ACM, 2009.
- [Snodgrass and Ontanon, 2014] Sam Snodgrass and Santiago Ontanon. A hierarchical approach to generating maps using markov chains. In *Tenth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2014.