

Interactive Genetic Engineering of Evolved Video Game Content

Erin J. Hastings and Kenneth O. Stanley
School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 32816
{hastings, kstanley}@eecs.ucf.edu

ABSTRACT

Procedural content generation techniques can increase replayability and lower the burden on developers by satisfying players' demand for new content. However, procedural content also creates an opportunity for new kinds of player-driven content customization by giving players access to the parameterized content space. This paper presents such a technique that enables players to manually customize evolved content represented by artificial neural networks. In particular, particle system weapons evolved by the multi-player space shooter called Galactic Arms Race (GAR) can be further "genetically engineered" by the players themselves in a new extension to the game called the *Weapons Lab*. Results are presented that demonstrate procedural weapons evolved by the game that are further customized by players in the *Weapons Lab*. The implication is that procedurally-generated content of many types can also be customized by players, adding a further dimension to its potential appeal.

Categories and Subject Descriptors

I.2.6 [Artificial Intelligence]: Learning - Connectionism and Neural Nets

General Terms

Algorithms

Keywords

procedural content, Galactic Arms Race, GAR, content-generating neuroevolution of augmenting topologies, cgNEAT

1. INTRODUCTION

Procedural content can help developers address the perpetual player demand for new game content. The idea in this paper is to provide additional value by enabling players to customize generated content analogously to customization options in games in which content is not procedurally

generated. This paper introduces such a method to allow players to customize procedural content evolved by artificial neural networks (ANNs), but the idea applies to other representations and content generation methods as well.

The platform for demonstrating this new technique is *Galactic Arms Race* (GAR)¹ [2, 3], a multi-player space shooter in which players fight with particle system weapons evolved by the *content-generating neuroevolution of augmenting topologies* (cgNEAT) algorithm. To explore player customization of the evolved weapons, GAR is extended with a new component called a *Weapons Lab*, in which players can manipulate evolved parameters and observe their effects on weapon dynamics immediately. The effect is a new kind of "genetic engineering" experience that allows players to directly explore the search space of parameterized content. While evolution in GAR is still important for the global search, players can effectively guide local searches that optimize weapon behavior to their preferences.

The main conclusion is that granting players access to the internal variables of procedurally-generated content is a further potential benefit of generative techniques that merits further exploration in future games.

2. BACKGROUND

This section begins by reviewing foundational work in procedural content generation and then focuses on Galactic Arms Race, which is extended to allow explicit customization in this paper.

2.1 Procedural Content Generation

Since the early days of interactive digital entertainment, games such as *Rogue*² and *NetHack*³ have featured randomized map and content placement. These games inspired an entire sub-genre of RPGs known as *Roguelikes* [8], which later directly influenced randomized map and item generation in more modern games, such as the popular *Diablo*⁴ series and *Dwarf Fortress*⁵, which features complex interaction among hundreds of different objects placed in a procedurally-generated game world.

While the path of content generation in these games is determined before the game starts, recent research has in-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

PCGames 2010 June 18, Monterey, CA, USA

Copyright 2010 ACM 978-1-4503-0023-0/10/06 ...\$10.00.

¹Multiplayer demo available at <http://gar.eecs.ucf.edu/>

²Copyright 1983 Artificial Intelligence Design

³Copyright 1983 Stichting Mathematisch Centrum

⁴Copyright 1996, 2000 Blizzard Entertainment, <http://blizzard.com/>

⁵Copyright 2002 Bay 12 Games, <http://bay12games.com/>

investigated enabling players to directly influence the path of content generation during the game. One such example is *Facade*⁶ [4], which generates interactive stories based on player choices throughout the game.

These contributions demonstrate successful content generation, but they do not attempt to discern whether the player enjoys the content generated, nor do they obviously extend to a shared multiplayer environment. Evolving procedural game content is an emerging research area that can potentially mold content to fit players' preferences by observing their behavior. Some of the few current examples of evolved game content include race tracks [6] and even the rules of the game itself [1, 7].

2.2 Galactic Arms Race

Inspired by the works in the previous section, the cgNEAT algorithm was introduced to evolve game content in real time, based on tracked player preferences in a multiplayer setting through a process called *collaborative content evolution* (CCE). This idea was first introduced in the Galactic Arms Race [2, 3] video game. In GAR (figure 1), the goal is to pilot a space ship to defeat enemies, gain experience, earn money, and most importantly, to find advantageous new weapons that are automatically generated by the cgNEAT algorithm [2]. GAR contains both a single player game and a full multiplayer game, in which weapons evolve based on the aggregate usage of all players online. In single player mode, evolution is directed by the actions of a single player battling NPC aliens in the game. GAR multiplayer evolution is substantially more diverse because the evolutionary population consists of the weapons currently possessed by all players in the game (i.e. it is CCE).

GAR's track record as a successful indie game makes it a good platform for further experimentation in content generation. Over 1,000 online players in GAR on the official server together killed over one million online enemies, mostly with evolved weapons [2]. The game appeared in Slashdot, won Best Paper Award at the IEEE Symposium on Computational Intelligence in Games (CIG 2009) [3], was editor's pick for Best AI in an Independent Game in 2009 at AIGameDev.com, and was also a finalist in the Indie Games Challenge⁷.

Each weapon in GAR is unique and represented by a compositional pattern-producing network (CPPN [5]), which is a type of artificial neural network (ANN). In GAR, players begin the game with an initial set of particle system weapons, which is the class of content evolved by the game. If players fire certain weapons often, cgNEAT infers that they enjoy that content, and the game produces new content that extends from or elaborates on that preferred content. In contrast, if players are unhappy with a certain weapon, they will not use it (or may discard it); thus the game will not produce more content of that type. The aim of this process is to continually evolve content based on the preferences of the players.

3. CUSTOMIZING PROCEDURAL WEAPONS IN GALACTIC ARMS RACE

⁶Copyright 2005 Procedural Arts, <http://proceduralarts.com/>

⁷See <http://www.indiegamechallenge.com/finalists/>



Figure 1: GAR Client. Players in GAR pilot their space ship (screen center) from a third-person perspective. This picture demonstrates a player destroying enemies with an evolved weapon. The GAR Client software is available online at <http://gar.eecs.ucf.edu> and runs on any Windows PC.

To customize an acquired evolved weapon in an extended version of GAR, players access the in-game *Weapons Lab* (figure 2), which presents a three-dimensional visualization of the evolved weapon CPPN and sliders to adjust its connection weights. Although the Weapons Lab is in effect a window onto the raw parameters of the underlying representation (which players may not fully understand), its interface is designed to look appealing and intuitive to gamers.

In particular, an enumerated list of connections (which are the parameters of the evolved CPPN) includes up and down arrows next to each entry that allow the player to increment or decrement parameter values without typing in numbers. The effects are visible immediately to the player, who can fire the modified gun in real time and observe its behavior. The hope is that this new ability to explore the space *around* popular evolved content will provoke renewed interest and curiosity in the game itself.

Several practical issues must also be considered when integrating customization with procedural content. The issues include (1) integrating customization into the game lore, (2) achieving the desired level of player control, and (3) limiting the frequency of customization. The right way to address these issues likely depends on the game; they are addressed in GAR as follows.

First, player customization is integrated into the game through the Weapons Lab accessible only at space stations. That way, customization is associated with specific *places* that thereby become important. The material that powers the weapons in the game is called *neuralium isotope*. When a player customizes a specific weapon, the actual CPPN graph is presented in a three-dimensional form reminiscent of an atomic isotope graph (figure 2, at left). In this manner, both the search for procedural weapons and customization of such weapons is integrated into the game's fiction.

Second, the goal of the Weapons Lab in GAR is to enable players to tweak their weapons, but not give players so much control that the search for new weapons becomes less meaningful. Thus, in GAR players may only adjust CPPN weights, but not add or remove connections (as evolution

can), nor change node activation functions. Thus there is still a need to explore the galaxy for new weapons no matter how much time is invested in tweaking weapons in the Weapons Lab.

Finally, to limit how often players may customize their weapon, they must trade in unused neuralium isotopes for the right to customize weapons. Thus, neuralium isotopes discovered in the galaxy that do not make good weapons nevertheless become a form of currency for purchasing modification rights, encouraging players to continue to search for neuralium even if they have no current desire for a new weapon. The hope is to demonstrate that customization can become an incentive for other activity in the game, thereby strengthening the argument for such interfaces in future games.

4. CUSTOMIZATION EXAMPLES

Figure 3 shows two examples of weapon modifications produced through the Weapon Lab. The leftmost weapon in both rows is an evolved weapon and modified versions from the Weapons Lab are to the right. The modifications usually preserve the overall theme of the weapon, that is, spread weapons remain spread weapons. However, the particular dynamics within that theme become malleable. Thus the player engages in a more local kind of search than the macro-level evolution that takes place in the overall game.

Interestingly, once these modifications are saved, they themselves enter the gene pool. Because multiplayer GAR mixes the genes of weapons held by different players online at the same time, other players may discover offspring of weapons manipulated through the Weapons Lab. That way, the Weapons Lab can influence the experience of players in the game even if they do not use it directly.

5. DISCUSSION AND FUTURE WORK

While this paper presents a preliminary snapshot of an idea with broader implications, it raises the possibility that manipulating the parameters of generated content can be fun. By exposing the player to the search space itself, the player in effect enjoys a learning experience about the structure of the search space. This experience can be rewarding or it can be frustrating depending on the nuances of the particular space, and the ultimate verdict will require more extensive testing and feedback from players themselves. The hope is that some players will demand customization to such a significant extent that they will be engaged in the game longer and willing to spend in-game currency for the right to such a privilege. Current results show that this idea is at least feasible in principle, and provide a proof of concept in which such custom genetic engineering is indeed integrated into an existing gaming platform.

The Weapons Lab is an ongoing project and will be released as part of a future release of GAR. Several possible improvements are under consideration. For example, an *undo* option would allow the player to revert to the original evolved weapon if desired. Also, it may be helpful to allow the player to change the parameter increment size to make searching easier. For example, the player might set a slider between “big changes” and “small changes.” Finally, the arrow buttons might be replaced with sliders to enhance the interface; however, the best interface likely depends on whether the platform is a joystick, keyboard, or something

else.

The main question for the future is whether this idea can be successfully applied to other types of content beyond weapon systems. Of course, the cgNEAT algorithm itself is today exclusive to GAR, but in the future if it evolves other classes of content in games, it comes with the added potential to enable in-game genetic engineering of evolved content. The idea can also apply to content generated through methods other than cgNEAT. Whether the content includes vehicles, clothing, or avatars themselves, the potential to let players navigate the search space is an intriguing possibility for games in the future.

6. CONCLUSIONS

This paper introduced the idea of in-game genetic engineering of evolved content. It was demonstrated in GAR in a new feature called the *Weapons Lab*. The result is that players can manipulate the underlying parameter space of evolved weapons in GAR. This ability is tied into the game and designed to enhance the players’ experience. The main contribution is to suggest the potential of the idea, and encourage its further exploration.

7. ACKNOWLEDGMENT

Special thanks to the Galactic Arms Race (GAR) development team, testers, and everyone who has downloaded the game since its release. The Galactic Arms Race free demo is available online at <http://gar.eecs.ucf.edu> and the project’s official email address is gar@eecs.ucf.edu.

8. REFERENCES

- [1] C. Browne. Automatic generation and evaluation of recombination games. *Ph.D. Dissertation*, 2008.
- [2] E. Hastings, R. Guha, and K. Stanley. Automatic content generation in the galactic arms race video game. *IEEE Transactions on Computational Intelligence and AI in Games*, 1(4), 2009.
- [3] E. J. Hastings, R. K. Guha, and K. O. Stanley. Evolving content in the galactic arms race video game. *Proceedings of the IEEE Symposium on Computational Intelligence and Games*, 2009.
- [4] M. Mateas and A. Stern. Procedural authorship: A case-study of the interactive drama façade. *Proceedings of the Digital Arts and Culture: Digital Experience: Design, Aesthetics, Practice*, 2005.
- [5] K. O. Stanley. Compositional pattern producing networks: A novel abstraction of development. *Genetic Programming and Evolvable Machines Special Issue on Developmental Systems*, 8(2):131–162, 2007.
- [6] J. Togelius, R. D. Nardi, and S. M. Lucas. Towards automatic personalised content creation for racing games. In *Proceedings of the IEEE Symposium on Computational Intelligence and Games*. IEEE Press, 2007.
- [7] J. Togelius and J. Schmidhuber. An experiment in automatic game design. In *Proceedings of the IEEE Symposium on Computational Intelligence and Games*. IEEE Press, 2008.
- [8] G. Wichman. A brief history of rogue. <http://www.wichman.org/roquehistory.html>, 1997.

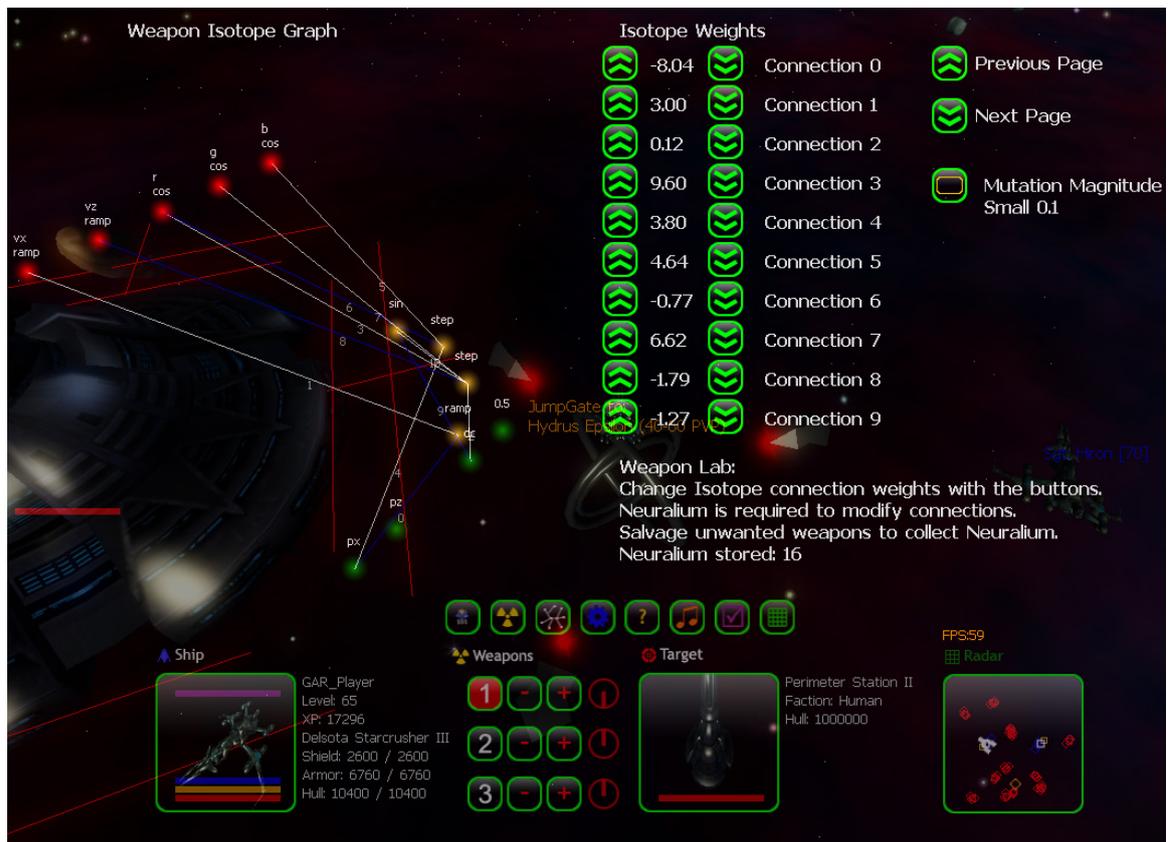


Figure 2: GAR Weapon Lab. A three-dimensional graph of a CPPN that encodes a particle weapon, referred to as a “neuralium isotope” in the game, is displayed on the left side of the screen. Arrow buttons that enable players to tweak connections weights are on the right-hand side. Players can modify the weights and immediately see the results by firing the weapon. That way, players can further customize already-evolved procedural content as they wish.

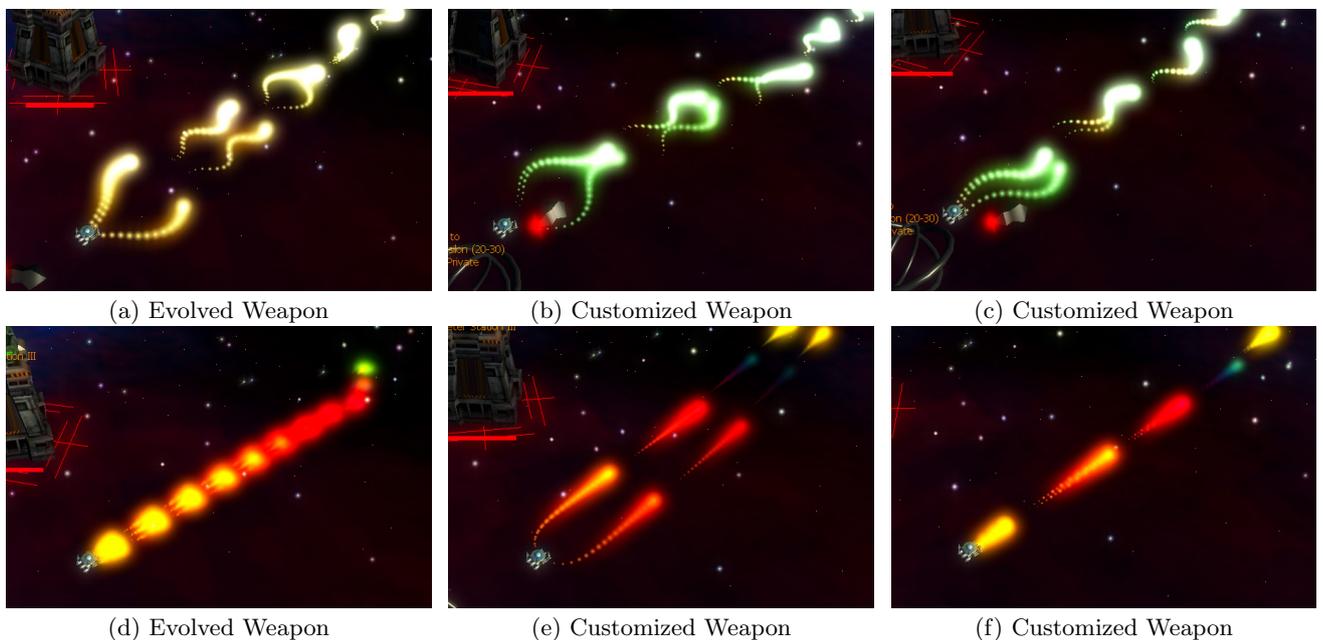


Figure 3: Evolved Weapons Customized in the Weapons Lab. Each row of three images displays a weapon automatically evolved by cgNEAT and two additional weapons customized from it by the player. These images give a sense of the potential of such customization.