Combining Search-Based Procedural Content Generation and Social Gaming in the Petalz Video Game

Sebastian Risi, Joel Lehman, David B. D'Ambrosio, Ryan Hall and Kenneth O. Stanley
Department of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 32816, USA
sebastian.risi@gmail.com, jlehman@eecs.ucf.edu, ddambro@eecs.ucf.edu, ryanhallcs@gmail.com, kstanley@eecs.ucf.edu

Abstract

Search-based procedural content generation methods allow video games to introduce new content continually, thereby engaging the player for a longer time while reducing the burden on developers. However, games so far have not explored the potential economic value of unique evolved artifacts. Building on this insight, this paper presents for the first time a Facebook game called *Petalz* in which players can share flowers they breed themselves with other players through a global marketplace. In particular, the market in this social game allows players to set the price of their evolved aesthetically-pleasing flowers in virtual currency. Furthermore, the transaction in which one player buys seeds from another creates a new social element that links the players in the transaction. The combination of unique user-generated content and social gaming in Petalz facilitates meaningful *collaboration* between users, positively influences the dynamics of the game, and opens new possibilities in digital entertainment.

Introduction

Search-based procedural content generation (PCG) is a recent topic of interest in video game research (Togelius et al. 2011) wherein parts of the game are created algorithmically rather than being hand-designed. The benefits of this approach are the reduced cost of designing new content and the unique experience each time the players enter the game. Evolutionary computation (EC) methods in particular have proven effective at automatically generating diverse media such as two-dimensional images (Secretan et al. 2011), three-dimensional forms (Clune and Lipson 2011b), musical compositions (Hoover et al. 2012) and even content for video games (Hastings, Guha, and Stanley 2009).

However, while EC methods facilitate the creation of new content, the potential for *economic value* of evolved artifacts remains unaddressed. While many current games (e.g. Diablo III[1], World of Warcraft[2], etc.) have virtual economies that essentially assign value to virtual goods, so far no game gives the user the ability to breed their own content that can be shared with other users through a global market. Such a market could offer players a continual stream of novel content that can keep them engaged longer and should also facilitate collaboration between different players. More fundamentally, the value created by genuine novelty suggests an intriguing unexploited opportunity to organize economic activity within a game around its own evolving content.

Figure 1: **Petalz Balcony View.** Players in Petalz can breed their own unique flowers and share them with others through a marketplace. This picture shows an example balcony that a player has decorated with various available flower pots and player-bred flowers. The offspring of a flower is shown as little seedlings, which give a preview of what the adult flowers would look like. Petalz is designed to be a casual and social Facebook game that is accessible to a large demographic. *Note: Pictures in this paper are best viewed in color.*

This paper introduces two novel technologies that take steps towards achieving this ambition: The first is a genetic encoding to represent aesthetically-pleasing flower images and shapes and the second is Petalz, a social video game on Facebook that demonstrates the creation of value through a marketplace of evolved content. The central game mechanic in Petalz is the ability for each player to maintain and breed a collection of unique flowers (Figure 1). The flowers in Petalz are genetically encoded by a special kind of compositional pattern producing network (CPPN; Stanley 2007). The special CPPN encoding enables the user literally to breed flow-

[1]Copyright 2012 Blizzard Entertainment, http://blizzard.com/
[2]Copyright 2004 Blizzard Entertainment, http://blizzard.com/

ers that share important features with their natural counterparts like symmetry, repetition, and distinct petals.

Analysis of Petalz market data collected during a four-month testing period suggests that the marketplace facilitates significant collaboration among players and that players generally prefer flowers descended from the marketplace. Additionally, by harnessing the fact that user-generated content has economic value, an explosion of credible content becomes possible. The main conclusion is that Petalz offers a new kind of gaming experience, combining social gaming with user-generated content, whose full potential is suggested by the initial exploration in this paper.

Background and Related Work

This section reviews related work on procedural content generation and also the technologies that enable the flower encoding introduced in this paper.

Procedural Content Generation (PCG)

PCG is a search-based approach wherein parts of the game (e.g. maps, textures, items, quests, etc.) are generated algorithmically rather than being hand-designed (Togelius et al. 2011). This approach can reduce design costs and also benefit players by presenting them with unique experiences each time they play. Many PCG approaches rely on a fixed set of parameters or random number generators, but recently focus has shifted towards applying artificial intelligence approaches to enhancing PCG.

Of particular note are EC and other meta-heuristic approaches (Togelius et al. 2011) that can further save on development costs and may be able to produce unique content beyond what human designers might have created. One popular technique is interactive evolutionary computation (IEC; Takagi 2001), in which the user guides evolution. One example is NeuroEvolving Robotic Operatives (NERO; Stanley, Bryant, and Miikkulainen 2005), in which players guide the evolution of a team of fighting robots. In Galatic Arms Race (GAR; Hastings, Guha, and Stanley 2009) weapons are evolved automatically based on user behavior. Avery et al. (2011) evolved several aspects of a tower defense game. In another example, Togelius and Schmidhuber (2008) experimented with evolving the actual rules of the game. The combination of user-generated content and multiplayer online games opens up an interesting avenue that games are just beginning to exploit.

As games become more social, interactions between players become more complex. Many games have virtual economies that assign value to virtual goods. This idea is most common in massive multiplayer online (MMO) games, in which users can find items that are typically generated from a predetermined set and then list them for sale to other players, usually in auction-type environments. This idea is taken even further in Diablo III[3], wherein users can sell the virtual goods they find for real money. The game Second Life[4] also allows users to sell virtual goods for real money (by means of exchanging in-game currency for real currency). Interestingly, the artifacts in Second Life are designed and created by the player, usually through significant effort. Petalz strikes a balance among these approaches by allowing players to sell flowers they bred themselves for virtual currency. That way, players can create new content without the need for initial expertise and still participate in the online economy. The next section explains the technology behind the flower encoding.

[3]Copyright 2012 Blizzard Entertainment, http://blizzard.com/

[4]Copyright 2003 Linden Lab, http://lindenlab.com/

Compositional Pattern Producing Networks

Compositional pattern producing networks (CPPNs) are a variation of artificial neural networks (ANNs) that differ in their set of activation functions and how they are applied (Stanley 2007). A special kind of CPPN described later encodes the flowers in Petalz. While CPPNs are similar to ANNs, the different terminology originated because CPPNs were introduced as pattern-generators rather than as controllers. This section explains the difference in implementation and application between CPPNs and ANNs.

While ANNs often contain only sigmoid or Gaussian activation functions, CPPNs can include both such functions and many others. The choice of CPPN functions can be biased toward specific patterns or regularities. For example, periodic functions such as sine produce segmented patterns with repetitions, while symmetric functions such as Gaussian produce symmetric patterns. Linear functions can be employed to produce patterns with straight lines. In this way, CPPN-based encodings can be biased toward desired types of patterns by carefully selecting the set of available activation functions.

Additionally, unlike typical ANNs, CPPNs are usually applied across a broad space of possible inputs so that they can represent a complete image or pattern. Because they are compositions of functions, CPPNs in effect encode patterns at infinite resolution and can be sampled at whatever resolution is desired.

Successful CPPN-based applications include Picbreeder (Secretan et al. 2011), in which users from around the Internet collaborate to evolve pictures, MaestroGenesis (Hoover et al. 2012), which evolves musical compositions, EndlessForms (Clune and Lipson 2011b), in which users can collaborate to evolve three-dimensional forms online, and the Galactic Arms Race (GAR) video game (Hastings, Guha, and Stanley 2009), which automatically generates particle system weapons. These examples demonstrate that CPPNs can evolve diverse content. The approach in this paper evolves the shape and coloring of flowers encoded by CPPNs with the NEAT algorithm, reviewed next.

Neuroevolution of Augmenting Topologies (NEAT)

NEAT begins with a population of simple neural networks or CPPNs and then *adds complexity* over generations by adding new nodes and connections through mutations. By evolving networks in this way, the topology of the network does not need to be known a priori; NEAT searches through increasingly complex networks to find a suitable level of complexity. For a complete overview of NEAT see Stanley and Miikkulainen (2002; 2004).

Most importantly, such complexification, which resembles how genes are added over the course of natural evolution, allows NEAT to establish high-level features early in evolution and then later elaborate on them. For evolving content, complexification means that content can become more elaborate and intricate over generations. In this paper, flowers encoded by CPPNs are evolved by NEAT. NEAT is chosen because (1) it is proven effective for evolving ANNs and CPPNs in a diversity of domains (Secretan et al. 2011; Hoover et al. 2012; Stanley and Miikkulainen 2002; Hastings, Guha, and Stanley 2009), and (2) it is fast enough to run in real time (Hastings, Guha, and Stanley 2009; Stanley, Bryant, and Miikkulainen 2005), which is required for an interactive system.

Generating Flower Images and Shapes

The purpose of an encoding is to define a *space* of possible content that can be explored. Although CPPNs have previously been applied to generating two-dimensional images (Secretan et al. 2011) and three-dimensional structures (Clune and Lipson 2011a) the same generic approach is not possible to apply when a *specific* class of image or structure is desired. Thus this section discusses the modifications made to the general CPPN representation to produce flower-like images. In short, the general idea behind the flower encoding in Petalz is to deform a circle such that the resulting shape resembles a flower.

The process begins by deforming the circle. Because this approaches focuses on a radial pattern, polar coordinates $\{\theta, r\}$ are input into the CPPN (Figure 2). For each value of θ, the deformed radius of the circle at that point (r_{max}) is queried by inputting $\{\theta, 0\}$ into the CPPN. Next, to fill in the colors of the flower's surface, each polar coordinate between 0 and r_{max} is queried with the same CPPN for a RGB color value. This approach produces a deformed, colored circle, but still may not produce flower-like images. Most natural flowers demonstrate basic radial symmetry in the form of their petals; this property is exploited in the encoding by inputting $sin(P\theta)$ into the CPPN *instead* of the raw θ value, which creates a repeating pattern of deformation and coloration. The optional P parameter allows control over the period of the sine function and thus the maximum number of repetitions around the circle.

Finally, to further improve the aesthetic of the flowers, the concept of *layers* is implemented to reflect that flowers generally have internal and external portions. To encode this property, a new flower is queried through the same CPPN for each layer L. Each such layer is scaled based on its depth and drawn on top of the previous layer. Thus the inputs to the CPPN are $\{\theta, 0, L\}$ and the outputs are $\{R, G, B, r_{max}\}$ to determine the shape of each layer (Figure 2). The internal coloring is then determined by querying the CPPN as with the outermost layer. Interestingly, because the layers are queried by the same CPPN, they are mathematically related, giving a natural look. Further optimizations on this general algorithm are employed to achieve the fast rendering necessary for an online game.

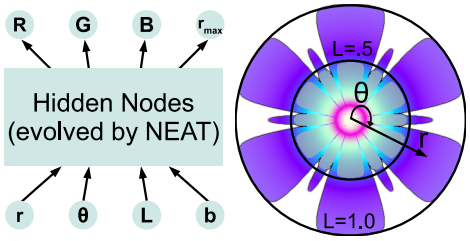


Figure 2: **CPPN Flower Encoding.** The CPPN that encodes flowers in Petalz takes polar coordinates (r and θ) as well as layer (L) and bias (b) values. The outputs are an RGB color value for that coordinate. The value r_{max} is also output, but only checked when $r = 0$ to determine the maximum radius for a given θ. The number and topology of hidden nodes is evolved by a standard CPPN-NEAT implementation (Stanley 2007).

The Petalz Video Game

This section introduces the Petalz video game (soon available publicly at http://apps.facebook.com/petalzgame/), which is the first game to implement a market for user-evolved content.

Development

Petalz is an online social game that currently runs through Flash on Facebook. The game is designed to be casual with an intuitive an easily accessible UI. The development of the game took over a year, by a six-member team including one artist. The Petalz Client contains 32,141 source lines of code (SLOC), the Petalz Server 3,422 SLOC.

Game Mechanics Overview

The central game mechanic for players in Petalz is the ability to maintain and breed a collection of unique flowers. All players possess a *balcony* (Figure 1), which they can decorate with various available flower pots. Players can also visit the balconies of their friends from Facebook (called *neighbors* in Petalz) and help their neighbors by watering or liking their flowers.

The primary interface through which players interact with flower evolution is a context menu that the player can open by clicking on a particular flower. The menu allows the player to (1) pollinate the flower, (2) cross-pollinate the flower with another flower, (3) post the flower to a friend's Facebook wall, (4) clone the flower, (5) inspect the flower's family tree, (6) sell the flower on the marketplace (described in the next section), or (7) store the flower in the player's inventory. Actions like pollination (i.e. mutation of a single flower genome), cross-pollination (i.e. crossover between two flower genomes; Figure 3) and cloning produce up to five little *flower seedlings*, which are miniature version of what the adult flower would look like. These seeds can then be planted by the user after which they take up to several hours to grow into their adult form.

Every flower that is not a clone is unique, which, as the results in this paper will show, allows players to continually

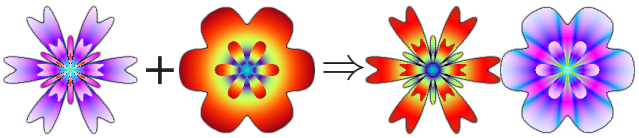


Figure 3: **Flower Cross-pollination.** By mating the CPPNs of two parents, children that exhibit the traits of both parents can be created. This approach gives players a powerful but intuitive way to explore the space of flowers.

Figure 4: **Petalz Marketplace.** The marketplace enables players to purchase flowers from others and to sell their own unique creations. Players can view the flowers listed by all players or by only a specific player, and can sort flowers based on their price, listing date or number of likes.

find novel flowers with characteristics evolved from flowers they preferred in the past.

The Flower Marketplace

The fact that each flower is a unique and lasting discovery opens up a number of intriguing possibilities. One is that players can sell their flowers in a global flower market to other players (Figure 4).

In Petalz the user can list a flower in the market either directly from her balcony or inventory. The listing price in the virtual "coin" currency, which users can earn by growing new flowers or performing other actions in the game, can be freely chosen by the seller in a range from 10 to 9,999 coins. Users have to pay 10% of the chosen price as a fee to list their flower for a maximum duration of six days. The listing fee is intended to discourage spamming the market with low-quality flowers.

Because flowers are genuinely novel artifacts, their seeds have potential economic value that can allow skillful breeders to be rewarded for their efforts. Furthermore, the transaction in which one player buys seeds from another creates a new kind of social interaction that links the players in the transaction. This interaction goes beyond simply enjoying the product of another player: Once seeds are purchased, the buyer can now breed new flowers from the purchased seeds, generating a whole new lineage.

The market in Petalz also has a section for pre-evolved *starter flowers*, from which the player can choose an aesthetically-pleasing flower to begin the game.

User Data	
Total Users	22
Max # of Flowers Evolved	1,206
Avg # of Flowers Evolved	155.14
Avg # Flowers Listed	37.54
Max # Flowers Listed	157
Avg # Flowers Bought/Sold	10.95
Max # Flowers Sold	36
Max # Flowers Bought	71

Table 1: **User Data.** A summary of a snapshot of user data records on the Petalz server taken on April 3, 2012 is shown. Users tended to evolve a significant number of flowers and interact often with the market.

Results and Market Analysis

Petalz was tested by 22 players (including the five authors) over a period of four months. A variety of statistics intended to illuminate the impact of the evolved flower marketplace on the Petalz game were derived from a data snapshot retrieved at the end of this period on April 3rd, 2012 from the game's database. Data tracking was initiated on December 1st, 2011. It is important to note that none of the extrema data, i.e. minimums or maximums in the tables in this paper, result from the authors.

Table 1 summarizes general information about the behavior of the players during the testing period. It is evident that players together yielded both significant evolutionary and market activity: The average player evolved no fewer than 155.14 flowers and listed on average 40 on the market.

Table 2 provides further insight into the dynamics of evolution in the game. First, over three thousand flowers were evolved by players suggesting that breeding is a significant attraction in the game. Furthermore, many flowers resulted from long evolutionary lineages for interactive evolution (103.19 generations on average separate flowers displayed on users' balconies from the flowers that seeded the game). In addition, data from "liked" flowers suggests that there is measurable variety in the aesthetic appeal of flowers and that users are more likely to display more appealing flowers on their balconies. Thus taken as a whole the data from Table 2 suggests that evolution within Petalz is driving force behind player behavior.

Given the significance of breeding within Petalz, an important question is what impact the evolved marketplace has on the game. To investigate this question, each transaction in the marketplace was recorded during the testing period. This recorded data is summarized in table 3. Some of these statistics establish that the marketplace is working intuitively. For example, cheaper flowers sold better than more expensive flowers, and not all listed flowers were purchased. Furthermore, the flowers that were bought tended to have

Flower Evolution Data	
Total Evolved Flowers	3,703
Total Liked Flowers	130
Max Generation	288
Avg Generation of Balcony Flowers	103.19
Avg Likes of All Evolved Flowers	0.04
Avg Likes of Balcony Flowers	0.5
Avg # of Users Contributing to Balcony Flower	7.27

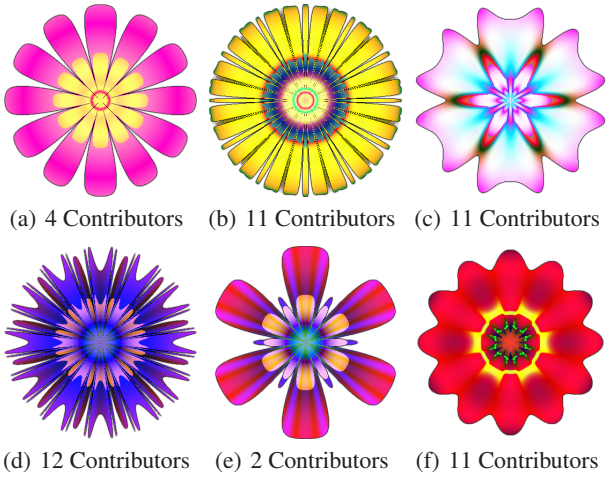
Table 2: **Flower Evolution Data.** Users collaborated indirectly to evolve many flowers and tended to display those found to be more aesthetically pleasing on their balconies.

Flower Market Data	
Total Market Listings	826
Total Market Sales	241
Avg Listing Price of Flowers	216.97
Avg Sales Price of Flowers	164.12
Avg Likes for Listed Flowers	0.36
Avg Likes for Sold Flowers	0.64
Avg # of Users Contributing to Market Flower	7.46
Proportion of Balcony Flowers Descended from Other Users' Flowers'	0.73
Avg Generations from Market to Balcony Flower	6.71
Total Currency Flow on Market	39,555

Table 3: **Flower Market Data.** Flowers that were cheaper or more aesthetically pleasing tended to sell better. Importantly, the market also facilitated collaboration between users.

more likes, indicating that flowers that appealed more aesthetically to users were also more highly *valued* by them. More significantly, the statistics also support that the marketplace is integral for both facilitating collaboration and evolving more desirable flowers. Importantly, at the time the data snapshot was taken, the majority of flowers displayed on users' balconies were close descendants of flowers bought from *other users* on the marketplace, although users can choose to ignore the marketplace completely or buy not from other users but from a permanent set of pre-evolved starter flowers that are always available on the marketplace.

Furthermore, marketplace flowers themselves were on average the product of many users collaborating, indicating that flowers bought from the marketplace were often further evolved and later sold to other users. Figures 5 and 6 present examples of this process. While Figure 5 shows a selection of user-evolved flowers, Figure 6 shows a phylogeny, collaboratively evolved by 13 different users. The tree reveals that Petalz together with its marketplace allows users to collaborate on a wide variety of flowers. While some mutations produce a minor change in the flower's phenotype (Figure 6a), other mutations produce a greater variety of different flowers that nevertheless share common features (Figure 6c). Cross-pollination allows players to breed flowers that exhibit a mixture of the traits of both parents (Figure 6b).



(a) 4 Contributors (b) 11 Contributors (c) 11 Contributors

(d) 12 Contributors (e) 2 Contributors (f) 11 Contributors

Figure 5: **Evolved Flower Examples.** The CPPN-based encoding allows the discovery of a great variety of aesthetically-pleasing flowers, which show varying degrees of complexity. The number of contributing users for each flower shows that the Petalz flower market facilitates meaningful collaboration.

Discussion and Future Work

The CPPN-based encoding in Petalz allows a wide variety of flowers to be discovered. Small changes in the genotype often produce small changes in the phenotype and therefore allow the user to meaningfully explore the flower search space. Because the underlying encoding can become more complex over time, users can potentially continually discover new and more complicated flowers. Also importantly, CPPNs in effect encode flowers at infinite resolution that can be sampled at whatever resolution is optimal.

The market in Petalz follows the insight that user-evolved artifacts introduce a new kind of economic value. Furthermore, the analysis of the market data suggests that the marketplace both facilitates meaningful collaboration between users and positively influences the game dynamics. Importantly, these new insights open up unexplored possibilities in social gaming. Purchasing a flower from another player creates a new social interaction that links the players in an ongoing collaboration. Additionally, players can post their flowers to friends' Facebook walls, a unique gift possible to grant even to those who are not Petalz players. These novel social interactions ultimately allow users to discover and experience a continual stream of new content beyond what the original artists and developers can provide.

However, the real long-term benefits of a market of evolved content will emerge from further research. The hope for a game that generates its own content like Petalz is to keep players engaged longer. Therefore in this context, it is an interesting question whether evolving flowers alone will prove a sufficient motivation for users to return to the game, or if the space of possible artifacts has to be further extended.

Ultimately, Petalz together with its market for player-evolved content takes a step towards a new type of so-

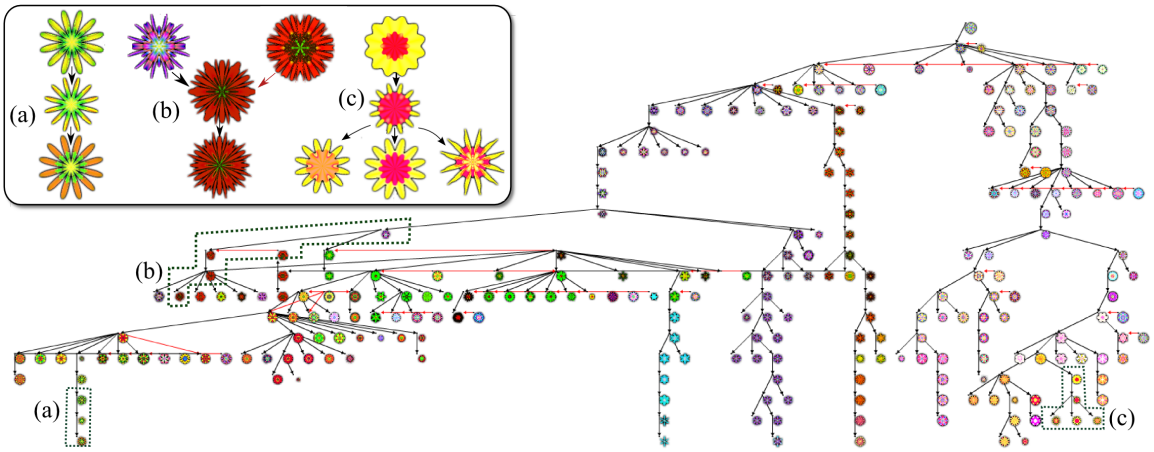


Figure 6: **A Phylogeny of Flowers.** This tree depicts the efforts of 13 different users. Each parent and child are separated by one generation (e.g. one pollination action). If a flower is the result of cross-pollination, the line connecting the second parent is shown in red. The inset shows modest phenotypic changes (a), a cross-pollination example (b), and more significant phenotypic changes (c) that nevertheless share common features.

cial game, where users can easily create new aesthetically-pleasing content and trade it with others.

Conclusion

This paper presented a novel genetic encoding for producing aesthetically-pleasing digital images of flowers and a social video game called Petalz based upon it. Based on the novel insight that evolved artifacts can create economic value, the game offers for the first time a global market of evolved content. Analysis of the market data suggests that the Petalz marketplace facilitates an online community where users collaborate to create and share their flower creations. For the game industry, Petalz raises new possibilities in social gaming; the continual stream of new content discovered through a shared global market could keep players engaged longer.

Acknowledgment

The flower encoding is patent pending under serial no. 61/541,967. Special thanks to the Petalz testers and Petalz artist Ashlee Hynes.

References

Avery, P.; Togelius, J.; Alistar, E.; and van Leeuwen, R. 2011. Computational intelligence and tower defence games. In *Evolutionary Computation (CEC), 2011 IEEE Congress on*, 1084–1091. IEEE.

Clune, J., and Lipson, H. 2011a. Evolving three-dimensional objects with a generative encoding inspired by developmental biology. In *Proceedings of the European Conference on Artificial Life, See http://EndlessForms. com*.

Clune, J., and Lipson, H. 2011b. Evolving 3d objects with a generative encoding inspired by developmental biology. In *Proceedings of the European Conference on Artificial Life (Alife-2011)*, volume 5, 2–12. New York, NY, USA: ACM.

Hastings, E.; Guha, R.; and Stanley, K. 2009. Evolving content in the galactic arms race video game. In *Computational Intelligence and Games, 2009. CIG 2009. IEEE Symposium on*, 241–248. IEEE.

Hoover, A. K.; Szerlip, P. A.; Norton, M. E.; Brindle, T. A.; Merritt, Z.; and Stanley, K. O. 2012. Generating a complete multipart musical composition from a single monophonic melody with functional scaffolding. In *Proceedings of the 3rd International Conference on Computational Creativity (ICCC-2012*.

Secretan, J.; Beato, N.; D'Ambrosio, D.; Rodriguez, A.; Campbell, A.; Folsom-Kovarik, J.; and Stanley, K. 2011. Picbreeder: A case study in collaborative evolutionary exploration of design space. *Evolutionary Computation* 19(3):373–403.

Stanley, K. O., and Miikkulainen, R. 2002. Evolving neural networks through augmenting topologies. *Evolutionary Computation* 10:99–127.

Stanley, K. O.; Bryant, B. D.; and Miikkulainen, R. 2005. Real-time neuroevolution in the NERO video game. *IEEE Transactions on Evolutionary Computation* 9(6):653–668.

Stanley, K. O. 2007. Compositional pattern producing networks: A novel abstraction of development. *Genetic Programming and Evolvable Machines Special Issue on Developmental Systems* 8(2):131–162.

Takagi, H. 2001. Interactive evolutionary computation: Fusion of the capacities of EC optimization and human evaluation. *Proceedings of the IEEE* 89(9):1275–1296.

Togelius, J., and Schmidhuber, J. 2008. An experiment in automatic game design. In *Computational Intelligence and Games, 2008. CIG'08. IEEE Symposium On*, 111–118. IEEE.

Togelius, J.; Yannakakis, G.; Stanley, K.; and Browne, C. 2011. Search-based procedural content generation: A taxonomy and survey. *Computational Intelligence and AI in Games, IEEE Transactions on* 3(3):172–186.