Title –

Name – Isaac Fagg

Student Number – S4920008

Abstract –

Contents/Figures -

**680/1400 Words**

**Introduction**

**Background/Context**

Procedural generation (PG) has had a place in the game industry for more than 40 years. Currently PG is used in a limited capacity in games used to help aid development by lessening work on certain aspects of a game. Games like Minecraft and No Man’s Sky use random PG instead of human-created content to create infinite levels for the player to experience. However, these worlds use the same formula infinitely to create the same level. A Minecraft plains biome near the spawn will be like the same biome 50,000 blocks away.

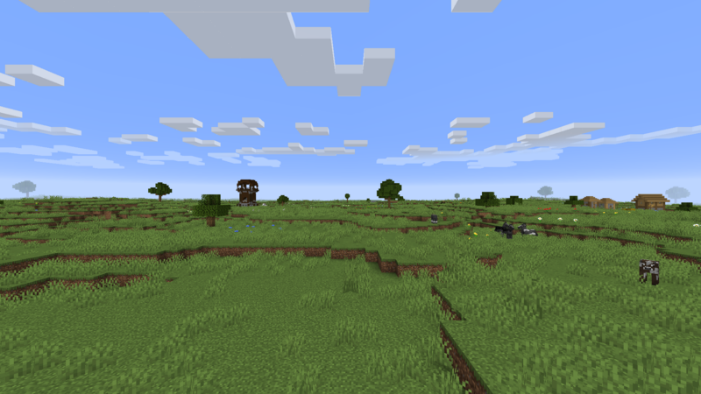


Figure - Minecraft Plains Biome (Lxazl5770 2020)

If instead the player was tracked to see what types of biomes they preferred, new biomes could be created that cater more to the player. A player who enjoys mountainous regions could see an increase of biomes focused on mountains

**Procedural Generation**

Offline vs online

**Search Based Procedural Generation**

Search-based procedural generation (SBPG) is a type of PG that relies on generating and testing content that fits certain criteria (Togelius et al. 2010, 2011). The tests performed do not just pass or fail content, instead they are assigned a fitness. A standard PG algorithm will only construct a single instance based on rules set by the designer. With SBPG multiple instances are created and then compared to attempt to generate the best content. While not essential, it is commonly used in conjunction with an evolutionary algorithm to perform the task of creating better content.

There are various approaches to SBPG, but the most popular are Predefined Evaluation, which has a set of established rules from which the content is tested against, and Interactive Evolution (Takagi 2001) where the user is the source of the testing by assigning a fitness based on the user’s selection or rating.

An example of Predefined Evaluation is a study by Browne (2008) which used SBPG to design rulesets for board games where the fitness based on the how the game performed. Some games did not use all the criteria available depending on the rulesets they had, all the criteria are shown in Figure 1. They found that measurements made during the evolutionary process were unreliable but proved useful for establishing if a game was viable.

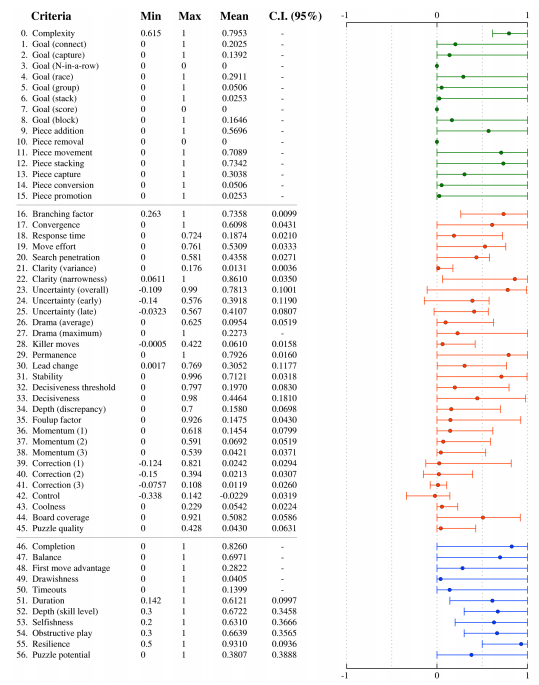


Figure - Criteria scores for all games. (Browne 2008)

A different study by Hastings in 2009 used Interactive Evolution SBPG to optimise weapons in a space shooter. Weapons were given a fitness based on how often they were used compared to the total time the player had access to them, with the most popular being used to create new weapons.

While these methods are popular, they are ill-equipped for personalised content. Using predefined fitness values does not allow much flexibility in what content is created, but it is useful in the cases of checking if content is viable. Interactive Evaluation suffers as content is only assigned a fitness when it is selected, which can lead to the content becoming similar.

**Adaptive Games**

Liapis et al. (2012) explored a new method of search-based procedural content generation, which changes the fitness function based on the player as shown in Figure 3. By selection content the entire fitness function is changed, so past content within the population is judged differently. This is done with the hope that as the player develops, their preferences do as well. The paper focused on presenting the potential of the method rather than proving its advantage over the other methods.

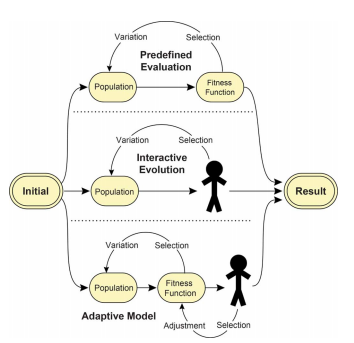


Figure 3 - Popular search-based procedural generation methods and the new Adaptive Model. (Liapis 2012)

**Content Evaluation**

For this project it was essential to have the ability to automatically assess the quality of game content. Liapis (2013) attempted to create a standard method for evaluating game content regardless of what content it is used on. In the study, various level types were analysed

Later Liapis (2015) explored using artificial intelligence trained to play levels like archetypical player behaviour

**Tracks**

There have been a variety of studies performed

A study by Cardamone et al. (2011) presented a framework for using Interactive Evolution for track design. Players would rate tracks after playing them and these tracks would be used to evolve future tracks. The population was shared with other users, so tracks produced were not tailored for the player, but it was shown to produce interesting and quality tracks. One of the tests they performed was how the rating system differed results. The player could either rank from 1 to 5 or like and dislike a track. Participants states they preferred the like/dislike interface as they felt they could express their preferences better, as they found they could not provide meaningful rankings with the rating system.

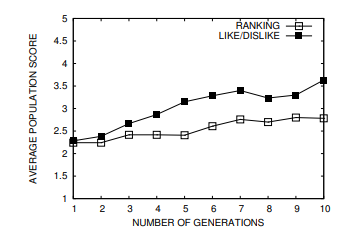


Figure - Population score using Ranking and Like/Dislike. (Cardamone et al. 2011)

**Player Experience**

Being able to identify what affects a player’s experience is critical to player-centered game design.

**Genetic Algorithms**

FI -2POP

Development & Implementation Report –

Self-Assessment of Learning

I managed to teach myself how to build an evolutionary algorithm

References