Title –

Name – Isaac Fagg

Student Number – S4920008

Abstract –

**25/1400 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 the quality of these levels are not always guaranteed.

**Procedural Generation**

**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 2010). 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.

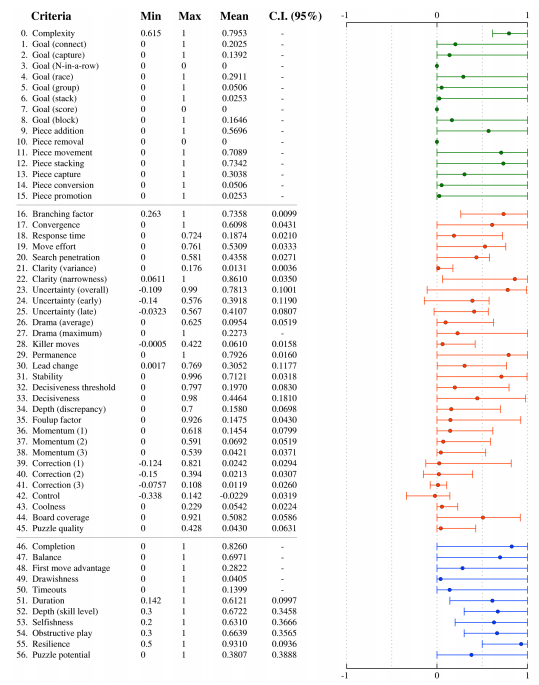


Figure - Criteria scores for all games

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 (2012) explored a new method of search-based procedural content generation, which changes the fitness function based on the player as shown in Figure 2. 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.

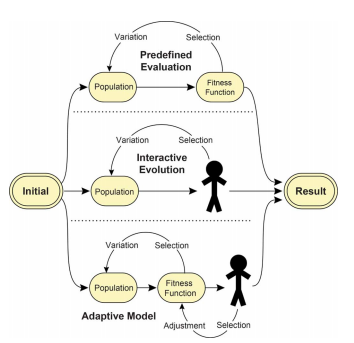


Figure 2 - Popular search-based procedural generation methods and the new Adaptive Model.

While the paper focused on visual aesthetics of content, the framework is applicable in other parts of content creation.

**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

**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

References