Dynamic Difficulty Adjustment with Limited Player Data

Game Difficulty

Traditionally the difficulty of a game is static, with just a single difficulty level (*e.g. Super Mario Bros.*), or multiple levels (*e.g. Halo 3*). But by analysing the player's performance, we can create a system that Dynamically adjusts the difficulty to best suit the player's skill.

Project Objective

To produce a framework for a Dynamic Difficulty Adjustment (DDA) system that is flexible enough to be applied to a variety of games. This framework makes use of the game developer's deep knowledge of their game, rather than relying on large amounts of player test data.

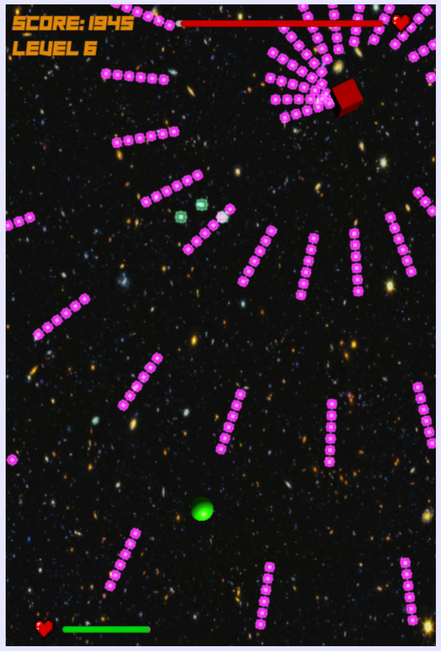

Figure 1: *A screenshot of the Implementation game*

Components of the Framework

The framework is made up of several modules:

Playstyle Functions: These functions compute how much the player is using each playstyle *(e.g. playing close to the enemy vs far away)*

Skill Functions: For each playstyle, these functions calculate how successfully the player is using each playstyle.

Counter Matrix: This matrix contains values representing the impact that each game variable or behaviour controlled by the DDA system has on difficulty of using each playstyle.

State Update Function: This function takes the outputs of the above modules and uses them to correctly adjust the values of each game variable and behaviour controlled by the DDA system.

Implementation

The Framework was implemented in a Bullet Hell game created in Unity3D, alongside an alternative system that used a static difficulty curve.

Results: Despite only two rounds of beta testing, the DDA system performed as well as the traditional difficulty curve of the placebo. It can be reasonably expected that with additional rounds of testing this system would surpass the static difficulty curve.

Difficulty Curve Erosion

The results of the implementation showed that *Difficulty Curve Erosion* occurred, where by repeatedly failing a challenge set by the game, the DDA will eventually reduce the challenge until the player can overcome it without ever needing to improve at the game, effectively *eroding* the Difficulty curve until it is flat. I extended the existing framework to prevent this erosion.
Initial testing of this extension on an artificial dataset showed promising results, but requires future research.

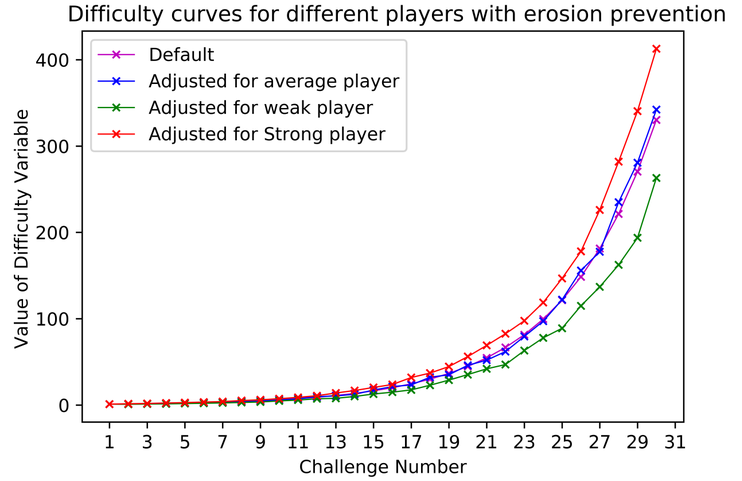

Figure 2: *A Graph showing the Difficulty Curve for different types of players*

Elliot Moffatt *University of Durham*
2nd May 2019