Electronics and Computer Science
Faculty of Physical and Applied Sciences
University of Southampton

Thomas A. E. Smith
taes1g09@ecs.soton.ac.uk
December 12, 2011

# Intelligent Procedural Content Generation for Computer Games

Project supervisor: E. Gerding - eg@ecs.soton.ac.uk
Second examiner: C. Cirstea - cc2@ecs.soton.ac.uk

A project progress report submitted for the award of
MEng Computer Science with Artificial Intelligence

**Abstract**

Increasingly, as the demand for ever larger and more varied computer game environments grows, procedural content generation (PCG) is used to ensure that content remains 'fresh'. However, many of the opportunities to use these systems to generated truly personalised content have so far been largely overlooked. When content is generated manually or algorithmically during the design phase of a game, it can only be created according to the designers expectations of the players needs. By instead generating content during the execution of the game, and using information about the player(s) as one of the systems inputs, PCG systems should be able to produce more dynamic experiences that can be far more tailored to enhance individual players' experiences than anything manually created. Much has been written about the generation of player models for the purposes of adaptivity or dynamic difficulty adjustment (DDA), and literature exists examining the problem of generating satisfying game environments via challenge adjustment. This project looks at combining these two fields to create an intelligent PCG system (IPCG) that is capable of monitoring players' progress and dynamically generating upcoming challenges to best suit their abilities.

# Contents

**1 Project Goals** **1**

**2 Project Background** **2**

**3 Proposed System** **4**

**4 Plan of remaining work** **5**

**Bibliography** **6**

# 1 Project Goals

# 2 Project Background

procedural content for many purposes (music strucure enemies textures) DDA resident evil use in existing games (valve, infintie mario) IPCG for difficulty, preference(fun), exploration

Procedural Content Generators have been used since the early days of gaming. Elite ¡one of the first great Y¿ had ¡stats¿, all procedurallu generated. These technique were required, as it was not possible to store the full data about that many unique planets on the distribution medai that was availabe at the time. As technologies improved; focus shifted more towards hand-crafted environments as it was easier to ensure that thes provided value and did not feel sparse [**?**]. However, with the further progress of technoloy attention has returned to Procedural generation. Modern game worlds contain vast amounts of detail, and procedural content generation algorithms are ideally suited to producing large numbers of variations on a theme [**?**], ¡expand¿ clouds textures sounds. Producing each of these items individually by hand would take many hours of labour and much disk space, but by definig specifivc sub elements and assmebly rules, variation can be almost endlessly reused. Not only used for cosmetics now either. Minecraft, infinite Mario?

Another facet of game design that has benefited ¡introduction to DDA¿ Previously, games had been limited to specific discrete difficulties as defined at production time. However, thes (can have been) considered to be overly restrictive - typically, if a games is begun with a certain difficulty it is difficult to later change; and this also aleinates players that are unfamiliar with the standerrs or uncertain how to classify themselves. Furthermore, since game difficulty is typically a continuous function of multiple parameters ¡stuff¿. Typically, DDA is achieved by altering values that are hidden from the player, such as enemy health, accuracy, or the amount of ammo and helathkits available in the world [**?**]. Often, the intention is to do this invisibly, and merely ensure that the player remains optimally challenged. By manipulating values behind the scenes, it is possible to ensure that the player is niether overchallenged, leading to frustration, or underchallenged, leading to boredom [**?**]. As DDA systems are given more controll over additional aspects of the game environment, they can begin to enter the realm of pcg, fundamentally(?) altering the stucture and pacing of the player's experinece. In the game Left 4 Dead, there is an 'AI Director' that is capable of estimating the ¡intensity¿ of each player's situation, and dynamically altering the generation of enimies of various types in order to ensure that the general flow of the game is kept exciting, ¡following build up, panic and calm phases¿ [**?**]. in Left 4 Dead 2, the director has additional control, adn is able to vary the stucture of the lever (something about the placement of ammo)(possible additnoal cite).

Typically, PCG is used in an offline manner, taking input only from a random seed. DDA, in contrast, uses inforamtion about the players' actions as an ipnput,

# 3 Proposed System

two parts adaptable PCG context free grammar wieghted following advantages in cite evaluator custom data list of possible data expect to initially be able to use k-means to disretise [1] [2]

# 4 Plan of remaining work

# Bibliography

[1] Robin Hunicke. The case for dynamic difficulty adjustment in games. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, ACE '05, pages 429–433, New York, NY, USA, 2005. ACM.

[2] Michael Nitsche, Calvin Ashmore, Will Hankinson, Rob Fitzpatrick, John Kelly, and Kurt Margenau. Designing procedural game spaces: A case study. October 2006.

And if you need to split it up use