

# Affectively Adaptive Level Generation for Mario

Peter Lehim Pedersen (pleh@itu.dk)  
Christoffer Holmgård Pedersen (holmgard@itu.dk)

Procedural Content Generation in Games, Fall 2012

**Abstract.** In this report, we describe the creation of a first emotionally adaptive level generation system for the Infinite Mario Framework. The system uses continuous physiological readings of sympathetic nervous activity (electrodermal activity and heart rate) as an expression of arousal from interacting with the game. These arousal values are then tied to individual level elements, using an artificial neural network to learn individual players' dispositions toward specific level design elements and use this to procedurally generate personalized levels, predict the player's response and evaluate the actual response to the generated content.

## 1 Introduction

What does it mean for a game to be fun? Enjoyable? Good, even? The subjective experience of emotions related to (video) gameplay are hard to define, quantify and measure objectively.

One possibility is to try and infer the player's engagement with the game, or aspects thereof, and use this as one measure of the quality of the play experience. Engagement can, in turn, with some reduction be operationalized into the player's emotional arousal; a unifying term for the relative sympathetic activation in an individual's nervous system at a particular time.

What in turn stimulates arousal in the player? One obvious answer would be the sum of events that take place over the course of a game, which in turn are unique to each particular game. For classic 2D platform games, the level of the game, in conjunction with the actions NPCs and the player, presents the primary predisposing configuration that determines which events will take place, and hence which level of arousal might be produced in the player.

For this project, we endeavored to create an adaptive level generator for the Infinite Mario Framework. The purpose of the adaptive level generator is to generate individualized levels that predispose the player's experience of a particular 'arousal curve', allowing a designer to predetermine the rise and fall of arousal over the course of a playthrough.

## 2 Background and state of the art

Arousal can be inferred from a multitude of behavioral and physiological signals - indeed the field of affective computing has it as a main goal to establish which

channels and to which degree. Two of the most well-established signals in terms of usefulness for measuring arousal are electrodermal activity and heart rate variability [REFERENCE PICARD HERE].

These physiological signals can be obtained from a game player with relatively little intrusion upon the player and provide responsive measures of arousal that can be gathered in real time [REFERENCE HERE].

Professional game development studios have successfully used electrodermal activity and heart rate variability as measures of arousal and used the signals to provide input to 'AI directors' that manipulate the events of a game to influence player arousal, but only in laboratory settings. A notable example is Valve's work on the Left 4 Dead 2 and Alien Swarm games, where in-game attacks from groups of enemies have successfully been timed to create a roller-coaster like experience of suspense.

Still, it would seem that the technology is not ready for deployment in commercial settings, even though some preliminary moves in that direction have been made by major console makers. In 2011 Sony several filed patents for controllers with embedded physiological measurement components and in 2009 Nintendo announced the development of the Wii Vitality sensor; a peripheral for the Wii console with sensors for electrodermal activity and heart rate variability. Still, none of these products have reached the consumer market to date. This may indicate that the integration of the use of psychophysiological signals into hardware, and perhaps game design, remains a difficult project outside of controlled laboratory conditions and that more research is needed to provide robust and useful methods for using these signals in ways that are relevant for game design. Recent research does, however, indicate that this is an avenue that may be worth pursuing [REFERENCES HECTOR AND GEORGIOS HERE].

Using the data for guiding the procedural selection and generation of content (including, but not limited to event scheduling) for enabling player-adaptive games represents one such opportunity, which we decided to explore for this project.

## 2.1 Game design

The Infinite Mario Framework, is a game derivative of the highly successful 2D sidescrolling platformer series, Super Mario Brothers. The game features a simplistic simulation of gravity that allows Mario to walk, run, and jump - and impressively move midair while in a jump! Using these abilities the object of the game is to guide the protagonist Mario through a level consisting of various obstacles and enemies.

All enemies exhibit simple, consistent behavioral patterns, bringing them close to being moving, deadly obstacles than NPCs proper. As such, the main determinant of the difficulty of any given play session is the configuration of the level.

Though the framework features a relatively small amount of obstacles and enemies to configure each level from, it allows for an practically infinite amount of variation though the placing of these elements and adjustement of the lengths of levels.

The original Super Mario Brothers games featured levels that were hand crafted by human level designers. This ensures a consistent experience for all players and graded difficulty curve throughout the game, but does not take individual player skill or preference into account.

Since every level is configured by individual components, or tiles, provides a prime opportunity for using procedural content generation to support or replace the human designer.

### **3 Methods**

In order to procedurally generate levels for Mario we needed a method for configuring individual tiles into levels in a way that would motivate gameplay with varying degrees of complexity, with the underlying assumption that this would lead to varying degrees of difficulty and hence engagement and arousal.

Though levels are fundamentally configured of tiles, the individual tile is not necessarily the best level for describing the challenges that the player faces in the game. A tile is quickly traversed by Mario, and the difficulty of doing so can only be determined by looking at neighboring tiles.

Therefore, we decided to understand particular configuration of tiles, or chunks of tiles, as the fundamental unit of challenges that the player faces in the game. The guiding principle for defining a chunk was to look at configurations of tiles that necessitate the player to perform one action, or one combination of actions, to traverse the configuration: for instance jumping over a gap, or onto an enemy to kill it.

Additionally, we defined the concept of screen-chunks: configurations of chunks that are close enough for the actions connected to the chunks to influence each other: E.g a jump over one chunk making Mario land near an enemy that he must kill or avoid.

The concept is illustrated in the Figure 1

#### **3.1 Results**

### **4 Discussion**

### **References**

