An Evaluation of Level-Based Procedural Generation in Movement-Based Video Games

Harry Findlay
School of Design and Informatics
Abertay University
DUNDEE, DD1 1HG, UK

ABSTRACT

Movement-based video games are reliant on a solid and consistent play space to work well. This usually takes the form of pre-built and pre-defined game worlds or maps, built and implemented by level designers. However, this research aims to investigate how gameplay mechanics in movement-based video games react to a procedurally generated level or environment. This research aims to seek the effects of an unpredictable and dynamic environment on movement-based gameplay mechanics.

Keywords

PCG – Procedural Content Generation
PLG – Procedural Level Generation
MB-VG – Movement-Based Video Games
WCF – Wave Collapse Function Algorithm
Gameplay Mechanics - GPM

Aim

This project aims to research the implementation of Procedural Level Generation (PLG) in Movement-based video games (MB-VG) and evaluate whether PLG is reliable and suitable enough to be used in future installments into the MB-VG genre.

Method

An algorithm will be developed (or tool used) to create a very simple procedurally generated level. All tools can be found via the epic store or unreal engine forums/documentation. There will be a focus on wall-like structures with a secondary focus on building-style structures, being more box-shaped. This environment would then be tested against a static environment and compared in certain areas of interest to the research. I then aim to have a group of individuals of no specific set of skills to test PLG against a static level and fill out a questionnaire.

Result

I expect that PLG would be a fun addition to MB-VG, however, would be too unstable in its generation to properly compliment the gameplay mechanics. I hypothesize that PLG would work for sub-genres of the MB-VG such as movement-based shooters (Titanfall 2, Respawn Entertainment, (2016)) but be less effective in core movement-based games such as Mirrors Edge (DICE, (2008)).

1. INTRODUCTION

Video games are one of the world's largest forms of digital entertainment where some people spend hundreds, if not thousands of hours playing the same game. However, on the contrary, some games barely keep people invested and entertained for ten or even twenty hours. For this reason, I aim to investigate and evaluate PLG and see what effect this will have on games with movement-heavy gameplay mechanics.

A massive part of playing video games is not only what goals or tasks the game has the player aiming for, but also where the player spends their time when doing so. Game worlds are one of the most important parts of a video game as this, not only, sets the scene, atmosphere, story and interactivity of the game, it can also represent what the player can do within the environment, how well it can be done and, in many cases, the limitations to the gameplay itself.

A common use of procedural generation in video games has it to either build play spaces or to populate a level with interactable of some sort. These interactable can be items, objectives, player-reactive factors in an environment and many video games use procedural generation for one or both factors. However, on the opposing argument, many video games of certain genres seem to neglect the use of PLG or procedural generation. Look at some of the world's leading game franchises such as Call of Duty Black Ops III (Activision, (2015)) or Titanfall 2 (Respawn Entertainment, (2016)), these games were high selling, fan favorite games with a high emphasis on movement-based gameplay mechanics, featuring player abilities such as sliding, dashing and wall running. These mechanics, however, are not only limited to these specific video games or their titles/series, but there are also hundreds, if not, thousands of games that adopt these game-play mechanics and game-style. Nonetheless, one thing that all these games seem to neglect is Procedural generation (PCG) and PLG.

My aim throughout this research project is to develop a series of movement-heavy gameplay mechanics and have players of varying skill levels test and evaluate these mechanics against a PGL and a static, pre-defined level and then voice their opinion on the experience that they had. My overall goal is to investigate and evaluate whether, in future, MB-VG should adopt PLG as a method of level and map development.

2. BACKGROUND

Procedural generation is a programming technique used since the 1980's in, not just video games, but in many areas of computer programming to create content on runtime.

Generally, PCG follows rules or guidelines when generating content so that it fits with the profile of the overall application or video game. Some strong examples of some of the video games that use PCG include: “Beneath Apple Manor” (The Software Factory, (1978)) and “Rogue”, (originally developed by Michael Toy and Glenn Wichman ((1980)).

PCG was originally developed to produce semi-random content for video games at run time. PCG was also used in other ways, some include a way to bypass system restrictions and to compress file sizes.

The modern uses for PCG are very different from the historic uses. In modern gaming, PCG is used to enhance the game in ways, rather than work around restrictions that plagued older computers. One example of PCG in modern video games is level generation, which is where this research will be focused. Levels, terrain, structures and other level-based topics can be semi-randomly generated on runtime to allow for replayability. Other areas of PCG includes procedurally generated stories, quests, and even game mechanics.

PCG comes with negatives as well, especially if poorly executed. Some PCG stories or story lines can lack depth and purpose. Some PCG levels can feel empty, lifeless and pointless. Many PCG creatures in games appear illogical, too random to make sense. On the contrary, if executed well, it can really make a game unique, look at No Man’s Sky (Hello Games, (2016)), where the PCG is so well implemented that the game contains 18 quintillion generated planets and planet-like structures.

2.1 Subsections

The Project Architecture

The PGL is the main obstacle and goal of this body of research, as the entire goal and hypothesis is built around the PCG and PLG. The main method of PCG in which this paper will be focusing is Wave Collapse Function Algorithm (WCF). WCF is an independent-minded algorithm which takes no external assistance or instruction, it simply receives input or prompt; the computer will then riff. Riff is the process where the computer takes the prompt, studies and disassembles the prompt, and then generates content of the same style – closely following the input given. The paper: Development of the system of procedural generation of game maps based on the wave function collapse algorithm written by Ilyin O**.** O**.** et al investigates the uses of WCF for PLG regarding building game maps. Within their research, see figures 3, 4, 5 and 6 for a visual representation of their map development. To discuss it here, they created and passed a series of map tiles into their WCF, where the algorithm took these tiles, preformed the riff. It was stated in the research that the tiles connectivity to one and other is important and needs to remain consistent across all tiles, this was the algorithm has an easier and more successful time analyzing and generating content from the given tiles or prompt. Throughout my research, I aim to apply the same processes when developing the PGL.

The Map Design

The design of the map will follow that of the popular MB-VG. The design of the level should contain a mix of open areas and narrow, tall hallways, bends and straight paths and overall horizontality and verticality. In order to better understand the guidelines in which the map design must follow, the type of player movement is needed to be known. The player will have the ability to simply walk around however, additionally, the player will have abnormal abilities. These movement abilities will include many different gameplay mechanics, which will include wall running, player dash, sliding and other movement of similar styles. These movement mechanics will be further expanded on in the next section. Researching more about parkour and the relationship it has with the environment, Julie Angel summarizes this fantastically and accurately in “Game Maps: Parkour Vision and Urban Relations” (Julie A, (2014)), where Julie researches and analyses the mindset of a parkour expert – and how this influences the unique connection parkour has with the environment in which they are in. Julie states that parkour experts are not limited by the environment surrounding them and instead, embraces it. “The wall is no longer ‘just’ a wall, the tree no longer ‘just’ a tree, the surface ‘just’ a surface.” (Julie a (2014)) (see figure 1). This uses the idea of Parkour Vision, an ability of re-perception where the person or player does not see objects as obstacles, but instead as tools, this will be a very important point that will be utilized when creating the tiles which will be used by the wave collapse function (WCF) in generating the map. Thus, when creating these tiles for the WCF, this mentality must be kept at forefront.

The Gameplay Mechanics

The gameplay mechanics (GPM) will be reminiscent of normal 3D games of both First Person and Third Person nature, as fundamentally the GPM behave the same way. Logically. For reference as to what to expect regarding GPM, a strong amount of research has been done into the already mentioned sources: Titanfall 2 (Respawn Entertainment, (2016)) and Mirrors Edge (DICE, (2008)). Some gameplay mechanics will include, but not be limited to: wall running, dashing, sliding and phasing (short range teleport). This section of the project will also make use of the paper,
Playful Mobility Choices: Motivating informed mobility dec ision making by applying game mechanics (Alexandra Millonig and Konstantin Mitgutsch (2014)).

2.2 Figures

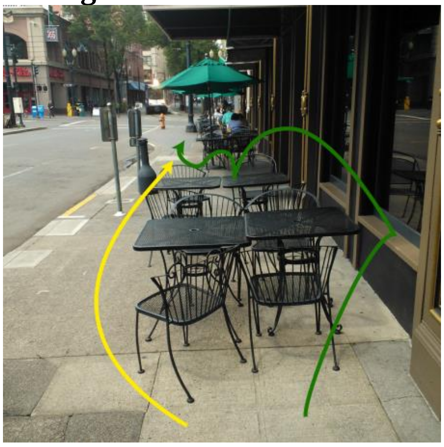

Figure 1 – A perspective explaining of ‘Parkour Vision’

3. METHOD

The methodology intended throughout this body of research is a mixture of game development ideologies. Such ideologies include: level or map design and development methods and guidelines, the development of various mobility-based gameplay mechanics and lastly, procedural map generation where the focus will be on the Wave Collapse Function Algorithm.

After some research, it was found that there are some guidelines which need to be followed when creating a successful and engaging play world. The largest and most important rule when developing any map or level is the flow of the level, and ensuring it fits the purpose of the game. In mobility or movement-based video games flow is particularly more important than that of non-movement-based video games. It was found that when the player is given advanced movement techniques, the game world need not be limited regarding what the player can do and where the player can go. The biggest addition when developing movement-based video game maps is verticality. Due to this, when creating the tiles which will be used as prompts for the wave collapse function algorithm, verticality will need to be implemented. This can be achieved by having some of the obstacles within the game world being higher than that of others. If done correctly, this will create a good flow as well as add versatility into the game world.

The gameplay mechanics need to support the map design and guidelines. The two main factors being flow and verticality, and so some gameplay mechanics have been researched and planned. The most crucial GPM that is requires is the wall running mechanic (Titanfall 2, (Respawn Entertainment, (2016))). Not only will this allow for flow form section to section throughout the level, but if developed and used correctly, the player will be able to gain height, thus, utilizing the verticality of the game world. Additionally, other GPM were researched and planned to help emphasize the high mobility of the player, such as the player dash as seen in DOOM ETERNAL (id Software, (2020)). This dash will allow the player to close larger gaps in platforms or walls, solidifying Julia Angel's research in Game Maps: Parkour Vision and Urban Relations (2014), where Julia talks about how walls are not 'just' walls and introduces us to the idea of Parkour Vision. With only two GPM we can see how these can influence and be influenced by the level and map design. This research will be investigating how we can use WCF within this formula and evaluate if the result will be a positive or negative reaction on the map and level development.

Expanding on this, WFC will be key throughout this research. The uses of WFC will be mainly to supply tiles, riff and generate a PGL for use alongside the GPM. The aim is to develop PGL using WFC to evaluate whether the user experience of movement-based gameplay mechanics is enhanced or diminished using PCG in level development. Upon researching the WFC, it was found that WFC is by far the most popular method of PLG in 3D video games. Specifically, the paper Development of the system of procedural generation of game maps based on the wave function collapse algorithm (Ilyan O.O. et al, (2024)), where the researchers created and supplied the WFC with pre-created tiles that contained roads, houses and trees and after the WFC's riff, the PGL was successfully created. This approach was studied and will heavily be influenced by the approach used within Ilyan O. O. et al's approach. The main difference between the research produced here and the research produced by Ilyan O. O. et al is the engine choice. I will be focusing on the use of Unreal Engine 5 whereas the referenced research focused on the use of Unity 3D.

4. Summary

WFC is a widely and commonly researched area of PCG in videos games. The unique topic in which this research aims to contribute to the WFC PCG field is: the relationship between advanced player mobility and its need for consistent and well flowing maps, and how WFC will influence the relationship. The aim is to develop a good enough tile set and algorithm that game developers and companies will consider using PLG in their development cycles. Not only will this speed up development times, this will allow programmers to help in areas of level design. Overall, the goal is to investigate and evaluate this research and conclude whether PLC can be used more in the future of movement-based video games.

5. REFERENCES

Angel, J.(2014). Game Maps: Parkour Vision and Urban Relations. In: Schiller, G., Rubidge, S. (eds) Choreographic Dwellings. New World Choreographies. Palgrave Macmillan, London. https://doi.org/10.1057/9781137385673_11

Büyükşar, O., Yıldız, D. and Demirci, S. (2024) *Enhancing wave function collapse algorithm for procedural map generation problem*, *Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi*. Available at: https://dergipark.org.tr/en/pub/ngumuh/issue/86158/1361413 (Accessed: 13 October 2024).

Heaton, R. (2018) *The wavefunction collapse algorithm explained very clearly*, *Robert Heaton*. Available at: https://robertheaton.com/2018/12/17/wavefunction-collapse-algorithm/ (Accessed: 13 October 2024).

Millonig, A. and Mitgutsch, K. (2014) 'Playful mobility choices: Motivating informed Mobility Decision making by applying game mechanics', *ICST Transactions on Ambient Systems*, 1(4). doi:10.4108/amsys.1.4.e3.

Ilyan O. O. et al. (2024) *Розробка системи процедурної генерації ігрових карт на базі алгоритму колапсу хвильової функції*, *Development of the system of procedural generation of game maps based on the wave function collapse algorithm*. Available at: https://journals.dut.edu.ua/index.php/sciencenotes/article/view/2948 (Accessed: 12 October 2024).

'PARKOUR VISION' (2013) *Headway Parkour Blog*. Headway Parkour Blog, 20 October. Available at: https://headwayparkour.weebly.com/blog/parkour-vision (Accessed: 13 October 2024).

Respawn Entertainment (2016) Titanfall 2 [Video game]
DICE (2008) Mirrors Edge [Video game]
Id Software (2020) DOOM ETERNAL [Video Game]