Procedural challenge generation guided by player choice in video games

Gonçalo Marques
goncalo.alexandre.marques@tecnico.ulisboa.pt

Thesis advisor: Carlos Martinho
Instituto Superior Técnico, Lisboa, Portugal

December 2020

Abstract

This work analyzes the feasibility of using procedural generation to create challenges in a video game based on the player's choices, such as weapon choice, and to compare that approach to one based on the player's skill as well as one based on generating content randomly. Few games have attempted to procedurally generate ways for the player to progress through the game, by generating challenges that keep the player learning new ways to use the existing mechanics. This work attempts to expand upon those concepts by three different ways of tailoring content to the player. We built a video game that generates content procedurally using the 3 aforementioned approaches and had several users test 3 different versions of the game, one for each approach. Our results suggest that, in this particular implementation, players preferred playing the random approach to the approaches with content procedurally generated, which leads us to believe that more work needs to be done to better understand how player adaptation needs to be implemented to improve play experience.

Keywords: Procedural Content Generation; Adaptive Content Generation; Challenges; Player Choice; Player Skill

1. Introduction

In this day and age, technology users expect to have content automatically curated to them, based on their preferences and consumption history. YouTube[1] recommends videos based on to the ones we are currently watching and have watched in the past while Spotify[2] plays us songs similar to the ones we liked. This is done in order to introduce new content that might be appealing to the user, which leads to increased engagement and consumption of the product or service.

In video games, PCG - Procedural Content Generation - is often used to generate content with the goal of increasing variety and therefore replayability and engagement. However, video games are very complex and generating a vast amount of engaging content can be challenging to do in a procedural manner: No Man's Sky[3], one of the most well known commercial games relying heavily on PCG, was criticized at launch by some players for generating content that, although vast, was also repetitive and unappealing[4].

Previous bodies of work have attempted to test the impact of procedurally generating content based on the player's skill, and concluded that it positively impacts the gameplay experience as opposed to not adapting content at all (see"Skill-based Progression Model for Smash Time" [1, 2] by João Catarino and "Holiday Knight: a Videogame with Skill-based Challenge Generation" [4] by João Pardal). With this in mind, this work's Research Question was the following: could it be that, between adapting content to the player's skill and adapting it to the player's progression choice, one of these approaches will have a more positive impact on the player experience? If so, which one and why?

2. Related Work

2.1. Procedural Content Generation

According to Togelius et al [6], PCG is the algorithmic creation of game content with limited or indirect user input. PCG methods are developed and used for a number of different reasons in games, including saving development time and costs, increasing replayability and allowing for adaptive games [3, 5]. Some examples of commercial video games that apply PCG are "No Man's Sky" and "Minecraft"[5]. Both of these games

[1] https://www.youtube.com
[2] https://www.spotify.com

[3] No Man's Sky (PlayStation 4, PC, 2016; Xbox One, 2018), an action-adventure survival game developed and published by Hello Games.
[4] https://en.wikipedia.org/wiki/No_Man%27s_Sky#Reception
[5] Minecraft (Windows, macOS, Linux, Android, iOS, 2011; XBbox 360, 2012; PlayStation 3, 2013; PlayStation 4, PlaySta-

were commercially successful and are able to generate vast amounts of content for the player to consume, despite having been made with a very small team of developers.

2.2. Challenge-Reward Dichotomy

When playing a video game, it is a common expectation that the player will receive a reward after completing a challenge. According to Wang and Sun [7], that reward can come in many forms, such as increased score and experience points, new in-game content, progression of the story (possibly through cut-scenes), or in-game items.

Wang and Sun also note the following:

- Rewards in video games, although mostly extrinsic[6], can provide intrinsic rewards[7] to the player. If an extrinsic reward (e.g. an achievement, an in-game item) is given to the player for mastering a specific mechanic or completing a level in a different way, it will motivate the player to learn more about the game's depth and possibilities, contributing to an intrinsically rewarding experience.

- There should be a balance between the effort/time spent trying to receive a reward and the value of the reward. If a player spends a long time trying to get a reward and then finally receives it, only to find out that it is much less valuable than what was expected, the player will be frustrated, which will increase the chances that he abandons the play session and, potentially, the game entirely.

Our takeaways from these points are that, with Adaptive PCG, we can achieve the following:

- Adapt content to the player's choices by trying to guide the player towards experimenting with different gameplay strategies, in order to create a more rewarding experience for the player.

- Measure the player's effort toward achieving an award: matching a reward's value with the player's effort to receive that reward is important, in order to make sure the player doesn't feel frustrated.

2.3. Adaptive Procedural Content Generation in Games

Adaptive Procedural Content Generation is a subtype of PCG that focuses on adapting the content generation to the player's behavior [6], which is the main approach we have used for this work.

MGS V[8] - Metal Gear Solid V - is an example of a game that adapts content to the player's choices. In it, you control a secret agent trying to infiltrate enemy facilities. You have access to lethal and non-lethal weapons and gadgets that you can use to dispose of or distract enemies. The game reacts to the player's play-style by modifying the challenges accordingly, for example:

- If the player has a preference towards infiltrating bases during the night, the game will add more enemies with night-vision goggles, who will have an improved vision cone compared to other enemies;

- If the player tries to dispose of enemies by shooting at them in the head, most enemies will start wearing helmets which nullify the first shot of the player.

Our adaptive content generation algorithm is based on an approach similar to MGS V's choice-based approach, where we adapted the challenges to the player's equipment choices, in order to make the game more or less challenging as needed.

2.4. Previous Skill-Based Work

Catarino and Martinho [1, 2] developed a skill-based progression model that attempts to address the problem of players with different skill levels tackling game content at the same pace, when in reality they might feel a different level of difficulty while facing the exact same challenges in a game. The hypothesis tested was whether generating challenges more suitable to a player's skill level would create a more engaging experience, which would increase play-time and enjoyment.

In order to test the hypothesis, an existing mobile game, "Smash Time", was modified. Smash Time has fast gameplay mechanics that result from the combination of elements from classic games like "Whack-a-Mole"[9] and "Space Invaders"[10], mixed with puzzle

tion Vita, Xbox One, 2014; Wii U, 2015; New 3DS, Switch, 2017), a survival adventure game developed and published by Mojang, later bought by Microsoft.

[6]Extrinsic reward: tangible or physically given to you for accomplishing something. It is a tangible recognition of one's endeavor: a certificate of accomplishment, a trophy or medal for winning a race, a monetary reward for doing your job. Because extrinsic rewards are tangible, they are usually given to the person doing the activity; as such, they are typically not from within the person.

[7]Intrinsic reward: an intangible award of recognition, a sense of achievement, or a conscious satisfaction: the knowledge that you did something right, or you helped someone and made their day better. Because intrinsic rewards are intangible, they usually arise from within the person who is doing the activity or behavior.

[8]Metal Gear Solid V: The Phantom Pain (PlayStation 3, PlayStation 4, Xbox 360, XBox One, PC, 2015), a stealth-action game developed by Kojima Productions and published by Konami.

[9]Whack-a-Mole, a popular arcade game that involves hitting plastic moles with a large, soft mallet, made in 1976 by Creative Engineering, Inc.

[10]Space Invaders (Arcade, Atari 2600, Atari 5200, Atari 8-bit, MSX, 1978), one of the first side-scrolling shooter games, developed and published by Taito.

mechanics.

Smash Time's modified game cycle, which incorporates the progression model developed by Catarino and Martinho, is comprised of the following steps:

1. Generate a new challenge (game content) to present to the player using:
   - the Player Performance Predictive System from the Player Performance Model;
   - the Content Variety Data from the Content Variety Model;
   - the Challenge Library.
2. Register the player response dealing with the obstacles that compose the generated challenge;
3. Analyze the player performance through the recorded player actions relative to the generated challenge;
4. Register the player performance data in the Player Performance Model;
5. Predict the player's performance in the Player Performance Predictive System;
6. Register the challenge variety data in the Content Variety Model;
7. Back to step 1.

The final evaluation of Catarino's hypothesis was made by having 2 groups of 16 players, one that tried the normal version of Smash Time, which increased difficulty equally across player skill levels, and one that tried the version with the aforementioned skill-based model, which would adapt the challenges to the player's skill level. The conclusion was that a skill-based progression model was able to increase not only the duration of play sessions, but also the number of play sessions per player. It was also concluded that this model has the potential to increase player immersion and, consequently, create more engaging gameplay experiences. All these positively contribute to the extension of a game's overall lifetime and revenue.

In a following body of work, Pardal and Martinho [4] built upon Catarino and Martinho's work on skill-based challenge generation by testing Catarino's hypothesis, how impactful it is to adapt the game's challenges to the player's skill as opposed to not adapting content at all, on a different style of game.

In order to test Catarino's hypothesis with a different type of game, a new video game, Holiday Knight (Fig. 1), was developed. Holiday Knight is heavily inspired by games like "Enter the Gungeon"[11] and "Binding of Isaac"[12] where the main character fights enemies by shooting at them while at the same time dodging their attacks (also known as "Shoot'em Up"[13] Games).

[11] Enter The Gungeon (Windows, OS X, Linux, PlayStation 4, 2016; Xbox One, Switch, 2017), a bullet-hell roguelike game developed by Dodge Roll and published by Devolver Digital.

Figure 1: A screenshot taken from Holiday Knight

In Holiday Knight, the player will go from room to room, defeating all the enemies in each room. When all the enemies have been defeated, the player can progress to the next room and repeat the process. When all the rooms have been cleared of enemies, the game ends. If the player dies during the process, the game also ends.

In order to test the aforementioned hypothesis, Pardal had a group of users play an adaptive version of the game, which would update the player's performance values according to his performance, and another group of users play a static version of the game, which wouldn't update the player's performance values and instead use the default values calculated in the preliminary evaluation. Both these versions would adapt the challenges to the performance values, but only one would update those performance values throughout the play session. It was concluded that, while the adaptive version showed no significant differences over the static version regarding the duration of the play session or the number of play sessions, players who tried the adaptive version reported feeling a more homogeneous level of challenge than the ones who tried the static version.

This work was built upon a modified version of Holiday Knight.

3. Solution

3.1. Approach

This work's research question, which PCG approach leads to a more pleasant player experience: content generation based on the player's skill or the player's choice, was tested using a modified version of the "Holiday Knight" video game described previously due

[12] Binding Of Isaac (Windows, OS X, Linux, 2011), an action roguelike game developed and published by Edmund McMillen and Florian Himsl.

[13] Shoot'em Up is a video game sub-genre of the shooter genre, where the player battles a number of enemies by shooting at them and dodging their fire, relying primarily on the player's reaction time.

to the fact that its gameplay and progression model have already been developed and tested in such a way that it can be built upon. The modified version of Holiday Knight has the following gameplay progression:

1. Similarly to the original version, the player must go from room to room, defeating all the enemies in each room, in order to progress to the next. This process will repeat until the player either reaches the last room or loses. In this version of the game, there are 20 rooms in total and when the player defeats all the enemies in the last room they win that playthrough.

2. The player can pick up weapons, which can be either assault rifles, shotguns or sniper rifles, that can be switched with other weapons that the player finds. Assault rifles have a 3-round burst fire and "2D Recoil" (the weapon rotates around the player as he shoots with it), shotguns fire two rounds at once in 2 different directions and sniper rifles are precise but have a slower rate of fire. Each weapon has an element associated with it that will cause different effects based on which element it is:

    - Fire - The enemy causes a burst of fire that damages enemies around it
    - Ice - The enemy freezes (i.e. doesn't move or attack) for a few seconds
    - Poison - The enemy's bullets will travel at half the speed

    These effects are applied to the enemies once they have suffered a certain amount of damage from the player's weapon.

3. Enemies have different levels of resistance against the effects suffered from the player's weapons. Some take less damage against fire, some already move slowly and hence are less affected by being frozen and some remain poisoned for a smaller period of time.

4. Each room also has a weapon placed in the corner, in order to allow the player to switch weapons throughout the playthrough.

3.2. Architecture

In order to support our modifications to the "Holiday Knight" game, we had to adapt the existing skill-based progression model. The conceptual differences between both models are shown in Fig 2, where white boxes are the common components between them, orange boxes are components related only to the skill-based progression model and blue boxes are components related only to the choice-based progression model.

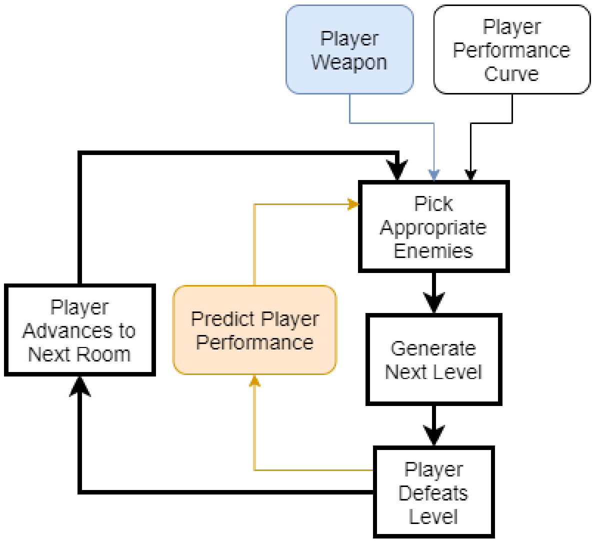

Figure 2: A diagram of both the skill-based and choice-based progression model.

The adapted skill-based model's event flow will work as following:

1. Using the Player Performance Curve and the Player Performance Prediction, several levels (rooms) will be randomly generated, and only the room with the best utility will be chosen as the player's next challenge, based on its enemies (further explained in section 3.3);

2. When the player defeats all the enemies in that room, the player's performance will be analyzed and his performance for the next level will be predicted based on previous values;

3. When the player advances to the next room, the cycle goes back to step 1;

The proposed choice-based model's event flow will work as following:

1. Using the Player Performance Curve and the player's current weapon, several levels (rooms) will be randomly generated, and only the room with the best utility will be chosen as the player's next challenge, based on its enemies (further explained in section 3.3);

2. When the player advances to the next room, the cycle goes back to step 1;

3.3. Progression models

In the following paragraphs, we will use the term "utility" as a way to describe how useful a certain component of the model is relative to the Player Performance curve's value in the current level. The enemies' utilities will mean how much their difficulty matches the current value of the Player Performance Curve. Performance is inversely proportional to difficulty, which means that a lower performance value will imply a higher difficulty and a higher performance value will imply a lower difficulty.

Both models will take the Player Performance Curve as input, which allow the game designer to specify the desired player performance throughout the game. It works similarly to a Difficulty Curve, but reversed: when the difficulty increases, the value on a Difficulty curve will go up, while the value on a Player Performance curve will go down.

The adapted skill-based model takes the player's performance into account, measured by the amount of time the player takes to defeat all the enemies in the room (the time that the player takes to defeat each enemy only starts counting when the player damages that enemy for the first time). This performance value is then used to predict the player's performance in the next level, and pick the most appropriate enemies according to that predicted performance value. It is important to note that the player performance values are guided by tags defined by the game designer (e.g. tags that define that enemy type like "Swampy", "IceZombie", etc.) and tags updated by the progression model (e.g. informative tags like "ShotsToKill", etc.).

Unlike the skill-based model, the choice-based model will take into account **not** the player's performance, but the player's equipment instead, which in the case of "Holiday Knight" boils down simply to the player's current weapon. As mentioned above, the player is able to pick up weapons, and their elemental modifiers will be taken into account when generating enemies for each room, as well as the type of weapon the player is currently holding.

Each time the player defeats all the enemies in a room and advances to the next room, the game randomly generates many rooms and picks the best one based on its utility (the higher the utility, the more appropriate that room is for the player). The choice-based progression model uses the player's equipment to pick the best room. Given that a room's content is comprised solely of enemies, its utility can be boiled down to its enemies' utilities, which are calculated by matching an enemy's difficulty value against the current value of the Player Performance Curve.

Because some rooms may have a weapon inside them (which isn't the weapon the player is carrying), in that case, the enemies' utilities take into account the average difficulties between the player's current weapon and the weapon that is already inside the room.

The enemies' difficulty values are based on the following:

- Player's weapon element - as mentioned above, all weapons have an element associated with them which dictate the type of effects the enemy will suffer. All enemies have different resistances to these effects:
  - Fire - Some enemies take less damage against fire while others take more damage.
  - Ice - Some enemies move slowly and also remain still for longer periods of time and are hence less affected by being frozen. Enemies that move very frequently are more affected by being frozen.
  - Poison - Some enemies are more resistant to poison and hence remain poisoned for smaller periods of time, while others are less resistant and remain poisoned for longer periods of time.
- Player's weapon type - enemies take different amounts of damage depending on their size and the weapon damaging them.
- Quantity - each room has between 2 and 4 enemies, and the more enemies a room has, the harder that room will be to complete.

As was mentioned in the previous paragraph, enemies with a high difficulty value are only chosen when the Player Performance curve value is low while enemies with a low difficulty value are only chosen when the Player Performance curve value is high. For instance: at the beginning of the game, the Player Performance curve's value will usually be high (in our approach, it was), which means that the game will choose enemies that are more fragile against the player's weapon element and easier to defeat with the player's weapon type. However, as the player's desired performance goes down, it will eventually reach a low enough value where the game will spawn enemies in greater quantity that will be more resistant to the player's weapon element and type.

In order to have a reference point to compare the skill-based and choice-based progression models against, we also added a 3rd, random progression model. This progression model will choose both the type and the number of enemies (between 2 and 4) in each room completely randomly, without taking into account the player's performance or their equipment.

4. Results & discussion

In order to answer our research question, we implemented the two aforementioned progression models (skill-based and choice-based) in the new version of Holiday Knight and then had players test those versions of the game. Before we began the Final Evaluation, we did two Preliminary Evaluations. Their description will be in the following sections.

4.1. Preliminary Evaluations

The first preliminary evaluation was done to test the overall structure of the activity, where we used the skill-based and choice-based progression models. We intended to know:

- Whether the users that did the activity had any questions about it
- That there was nothing stopping them from starting or concluding the activity
- That they enjoyed the game enough to be engaged with it
- Whether there were distinct signs of their experience that differed along with the progression model assigned to them.

Because the gameplay experience would be similar to that of the original game, we decided to keep the bootstrap values for the enemies' performance in the skill-based progression model as-is, which were taken from Pardal's original preliminary evaluation.

Users started out by answering a short survey, where they were asked a few questions related to demographics, such as their age and how often they play video games. At the beginning of the survey, they were also told that the only difference between the two progression models they tested were how the enemies were chosen for each room. After answering these questions, they were asked to launch the game.

When players launched the game, a Player ID was automatically attributed, as well as which of the two progression models (choice-based or skill-based) they would be assigned first. In order to manage both these properties, we set up a Node.js server on Microsoft's Azure cloud platform, which would keep track of the last issued Player ID and progression model.

Players would go through a short tutorial where they were able to experiment with the game's mechanics at their own pace. After that, each player had to complete 2 playthroughs of the game, one using the skill-based progression model and another using the choice-based progression model, each consisting of 15 rooms. After they completed each of their playthroughs, the game would ask them to go back to the survey and answer questions related to their gameplay experience.

These are the stats that we tracked during each playthrough for this study, which were sent to a Node.js server after the player completed his playthrough:

- The specific types of enemies defeated, in which room they were defeated and how long it took the player to defeat them;
- Which progression model was used in that playthrough;
- How much time the player spent in the tutorial room, before starting his first playthrough;
- How much time the player spent in his playthrough;
- How many rooms the player was able to complete in each of his playthroughs;
- The weapon the player used to defeat the last enemy in each room;
- How much health the player had left after completing each room.

These stats were tracked both in the preliminary and final evaluations. We will be reporting the mean and median of some of these stats in the following section, in the context of the final evaluation. In order for us to receive these stats, the game would automatically send the gathered stats to the aforementioned Node.js server after the player finished his playthrough.

After doing this activity, we realized that it would be best if we also had a third progression model, that would simply generate enemies randomly, to serve as a baseline. This caused us to do several modifications to the game used for the activity:

- Because we decided to have a 3rd progression model, we thought the best alternative would be not to have each player play all 3 progression models, but rather to have each player play only 1 of them. With this in mind, players would now play only 1 of the progression models, which would be picked sequentially every time a new player would start playing the game.
- Because players would now only do 1 playthrough, we raised the number of rooms to beat the playthrough to 20 and also the minimum rooms needed in order to finish the activity, in total, to 20. Based on the preliminary evaluation, this would be a good balance between allowing the player to have enough time to experience the game as well as not having an activity that would be too long and potentially become annoying.

After making these modifications, we began the second preliminary evaluation, in which participated a total of 6 players. After doing so, we had mixed results about which progression model players had a more positive experience with, however, we believed that there was mostly a preference toward one of the non-random progression models compared to the random progression model: the lowest value for the competence, flow, challenge and positive affect dimensions, as well as the highest value for the tension/annoyance and negative affect dimensions, were all reported by either one of the two players that had played the random progression model. These results led us to believe that, although there wasn't a clear preference between the two non-random progression models, that there was a higher preference for those compared to the random progression model. As such, it made sense to proceed with the final evaluation.

Table 1: Gameplay Experience statistics for players who were assigned the skill-based progression model in the second preliminary evaluation.

	Median	Maximum	Minimum
Competence	13	15	11
Flow	16.5	17	16
Tension/Anoyance	2	4	0
Challenge	12.5	13	12
Negative Affect	0	0	0
Positive Affect	10	11	9

Table 2: Gameplay Experience statistics for players who were assigned the choice-based progression model in the second preliminary evaluation.

	Median	Maximum	Minimum
Competence	9	12	6
Flow	13	15	11
Tension/Anoyance	1.5	3	0
Challenge	10.5	13	8
Negative Affect	0.5	1	0
Positive Affect	8	8	8

Table 3: Gameplay Experience statistics for players who were assigned the random progression model in the second preliminary evaluation.

	Median	Maximum	Minimum
Competence	8.5	14	3
Flow	13	18	8
Tension/Anoyance	4.5	9	0
Challenge	6	8	4
Negative Affect	1.5	2	1
Positive Affect	6.5	9	14

4.2. Final Evaluation

For the final evaluation, we kept the modifications done for the second preliminary evaluation. We ended up gathering 20 complete results for both the skill and choice-based progression models and 25 for the random progression model. Some of the results we gathered weren't complete, due to the fact that some players didn't finish their playthrough of the game before they submitted their survey. An important thing to note is that there is a difference between finishing the game, which was optional and involved completing 20 rooms without being defeated, and completing their playthrough, which was necessary and involved completing 20 rooms in total. The main thing to note about these results is that we know that these players played the game and answered the survey, we just couldn't gather any of their gameplay stats. In this chapter, we will refer to these incomplete results as players who didn't submit their gameplay data.

Keep in mind that all the tests were done remotely, where each user would independently answer the survey and play the game on their own machine, without any supervision on our side.

4.3. Results

Due to the fact that, when testing different subsets of the data, not all of the 6 dimensions of the GEQ followed a normal distribution, we did a combination of parametric and non-parametric tests, verifying the normality of said subsets whenever applicable.

The results we got showed that:

- While only taking into account players who considered themselves familiar with similar games, players who played the random progression model felt more positive affect that those who played the choice-based progression model.
- While only taking into account players who had completed more than the median amount of rooms, players who played the random progression model felt less tension/annoyance that those who played the skill-based progression model and players who played the random progression model felt more competent than those who played the skill-based progression model.
- Players who didn't use the Ice Rifle the most felt more positive affect than those who did and players who used the Ice Rifle the most felt more tension/annoyance than those who didn't.

These results suggest that players preferred the random progression model over both the skill and choice-based progression models. A possible explanation for these results is that the difference in enemy quantity and type will be less pronounced in the non-random progression models, given the gradual increase in difficulty, which may cause the content to feel rather similar from room to room. We had removed the Content Variety model originally present in Holiday Knight in order to minimize the differences between the skill-based and choice-based progression models, given that the choice-based progression model did not make use of the Content Variety model. This is likely something that should be reconsidered in future work, where the Content Variety model should be restored to the skill-based progression model, and possibly added to the choice-based progression model as well, given that it may be an important aspect of the gameplay experience for this experiment. Another reason may be that the loot presented to the player isn't of enough value to serve as a strong foundation for the choice-based progression model.

They results also suggest that players, in general, didn't enjoy using the Ice Rifle, which may be due to the fact that it plays differently from the other weapon types, particularly the "2D recoil" and burst-fire features.

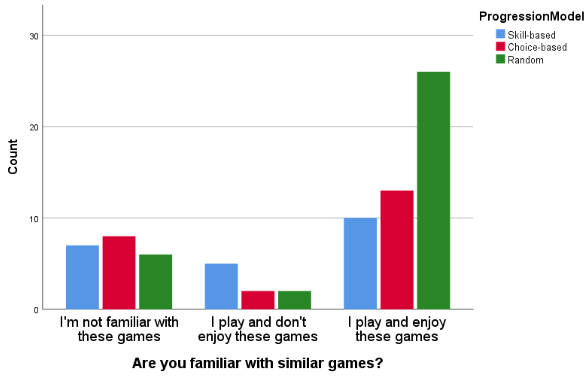

Figure 3

There was a variety of different weapons that players felt were most valuable to them, which leads us to believe that there wasn't a specific weapon that was clearly more valuable/powerful than the other weapons. We consider this a good thing because, if there was a universal unanimity between players as to which weapon was more powerful, that would mean that that weapon would probably be unbalanced compared to all the others. Because this unanimity wasn't present, we believe the weapons to be relatively well-balanced in general.

It's possible that the disparity between the number of players who played the random progression model compared to the players who played the non-random progression models may have influenced the results in one way or another. When separating players between three groups, one for each progression model, we found that the players who played the random progression model were much more likely to be familiar with and enjoy games similar to Holiday Knight, as is shown in Fig. 3.

Due to the COVID-19 pandemic, doing offline experiments with players in an office or lab was not feasible, so we resorted to doing online, unsupervised experiments, so that we could more easily achieve the amount of results we wanted, which caused, among other things, players who didn't complete the game the way we had intended. A specific case we encountered was that of a player who spent over 10 hours in the tutorial, but then spent very close to the mean amount of time completing the playthrough, which may have been due to the fact that he left the game running for many hours in the tutorial while doing something else before returning and completing the experiment.

5. Conclusions

This work's goal was answering the following research question: which PCG approach leads to a more pleasant player experience: content generation based on the player's skill or the player's choice. In order to answer this question, we adapted an existing game, Holiday Knight, and asked players to play it while measuring their gameplay experience in different versions of the game. The original version of Holiday Knight contained only a skill-based progression model, to which we added a new choice-based progression model. The way in which both of these differed was simply in the way in which they chose which enemies to present to the player at any given time: the skill-based progression model would choose these enemies based on the player's past performance facing them, while the choice-based progression model would choose them based on the weapon the player was currently carrying. In order to have a baseline progression model to compare them to, we added a third progression model that would simply choose both the amount of enemies and the individual enemies at random for the player to face.

In the experiments we did, we were not able to arrive to a definite conclusion regarding our research question. There was a slight preference, at best, toward the skill-based progression model compared to the choice-based progression model, however, there was a clear preference for the random progression model when compared to the other progression models. Several factors may have contributed to this outcome, and we've presented a set of guidelines to help with future follow-up work.

In future work, a greater emphasis should be put on making sure the content presented to the player is varied and that the loot presented is of value to the player. We had decided to remove the Content Variety model from Pardal's original version of Holiday Knight, in order to have the skill and choice-based progression models differ as little as possible, but it is likely that this will have to be reconsidered for future work.

We also suggest that the weapon types and elements that players have access to at the beginning of the game should be randomized, so that the bias the players may have in choosing the closest weapon to them doesn't affect the use of any one weapon combination in specific over the others.

The COVID-19 pandemic limited our possibilities when it came to offline experiments with testers, so we had to come up with an alternative way of doing those experiments. We chose to do offline, deferred, unsupervised experiments, given that players shouldn't need any guidance during the process, which caused us to have difficulty interpreting some outliers in our dataset, as well as not having received the complete results from every player. In the future, similar experiments should be more cautious about this approach, and consider doing online, supervised experiments instead, although that may slow the process of gathering results considerably.

References

[1] J. Catarino and C. Martinho. Skill-based progression model for smash time. In *Proceedings of IEEE Conference on Games 2019*, 2018.

[2] J. Catarino and C. Martinho. Skill-based progression model for smash time. Master's thesis, Instituto Superior Técnico, 2018.

[3] M. Hendrikx, S. Meijer, J. V. D. Velden, and A. Iosup. Procedural content generation for games: A survey. *ACM Trans. Multimedia Comput. Commun. Appl.*, 9(1), 2013.

[4] J. Pardal. Holiday knight: a videogame with skill-based challenge generation. Master's thesis, Instituto Superior Técnico, 2019.

[5] G. Smith, E. Gan, A. Othenin-Girard, and J. Whitehead. Pcg-based game design: Enabling new play experiences through procedural content generation. In *Proceedings of the 2Nd International Workshop on Procedural Content Generation in Games*. ACM, 2011.

[6] J. Togelius, N. Shaker, and M. J. Nelson. *Procedural Content Generation in Games: A Textbook and an Overview of Current Research*. Springer, 2016.

[7] H. Wang and C.-T. Sun. Game reward systems: Gaming experiences and social meanings. In *Proceedings of the 2011 DiGRA International Conference: Think Design Play*, 2011.