Universidade Estadual de Londrina

LUIZ ANTONIO LIMA RODRIGUES

ASSESSING THE INFLUENCES OF PROCEDURAL LEVEL GENERATION THROUGH A DIGITAL MATH GAME: AN EMPIRICAL ANALYSIS

Londrina
2018

LUIZ ANTONIO LIMA RODRIGUES

ASSESSING THE INFLUENCES OF PROCEDURAL LEVEL GENERATION THROUGH A DIGITAL MATH GAME: AN EMPIRICAL ANALYSIS

Dissertation presented to the Master's Program in Computer Science at Londrina State University to obtain the degree of Master in Computer Science.

Advisor: Prof. Dr. Jacques Duílio Brancher

Londrina
2017

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UEL

Rodrigues, Luiz Antonio Lima.
Assessing the influences of Procedural Level Generation Through a Digital Math Game: An empirical Analysis / Luiz Antonio Lima Rodrigues. - Londrina, 2018.
74 f. : il.

Orientador: Jacques Duílio Brancher.
Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Centro de Ciências Exatas, , 2018.
Inclui bibliografia.

1. Procedural Content Generation - Tese. 2. Player Experience - Tese. 3. A/B test - Tese. 4. Educational Game - Tese. I. Brancher, Jacques Duílio. II. Universidade Estadual de Londrina. Centro de Ciências Exatas. . III. Título.

LUIZ ANTONIO LIMA RODRIGUES

ASSESSING THE INFLUENCES OF PROCEDURAL LEVEL GENERATION THROUGH A DIGITAL MATH GAME: AN EMPIRICAL ANALYSIS

Dissertation presented to the Master's Program in Computer Science at Londrina State University to obtain the degree of Master in Computer Science.

EXAMINATION BOARD

Advisor: Prof. Dr. Jacques Duílio Brancher
Londrina State University

Prof. Dra. Cinthyan Renata Sachs
Camerlengo de Barbosa
Londrina State University

Prof. Dr. Eduardo Henrique da Silva
Aranha
Federal University of Rio Grande do Norte

Londrina, December 17, 2018.

ACKNOWLEDGEMENTS

I'm thankful to God, my family, professors, friends and to everyone that collaborated in this process. Furthermore, this study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

RODRIGUES, L. A. L.. **Assessing the influences of Procedural Level Generation Through a Digital Math Game: An empirical Analysis**. 2018. 74p. Master's Thesis (Master in Computer Science) – Londrina State University, Londrina, 2018.

ABSTRACT

Game development is an expensive process which commonly requires a multidisciplinary team. Procedural Content Generation (PCG) can remedy some problems of this process, aiding on the creation of different types of contents (e.g. levels and graphics), in both development and playing time. However, little research has been done in terms of how PCG influences players, especially on Digital Math Games (DMG). This dissertation tackles this problem by investigating how PCG influences players of a DMG. To this end, an A/B test was performed wherein the only difference was that one version (*static*) had human-designed levels whereas the other (*dynamic*) provided procedurally generated levels. To validate it, a two samples experiment was designed where each sample played a single version. Thereafter, in-game and questionnaire data from a total of 724 players were gathered and empirically analyzed. Based on it, this work delivers four main contributions: (1) a game that is evaluated as fun, encourages players to practice math and arise their curiosity and willingness to play it again; (2) showing that, according to questionnaire data, the experience of players from the *dynamic* version were similar to the ones of the *static* in all but one question, while being more difficult and providing equivalent engagement; (3) empirical evidence that players curiosity has a strong correlation to experienced fun and willingness to play the game again; and (4) evidence that players demographics and in-game performance have small to moderate impact on their experiences. We discuss these findings concerning to from where those differences emerged and how they can collaborate with game development, player modeling and on the game's use on educational ends.

Keywords: Procedural Content Generation. Player Experience. A/B test. Serious Game. Educational Game.

RODRIGUES, L. A. L.. **Avaliando as Influências da Geração Procedural de Fases Através de um Jogo Digital Matemático: Uma Análise Empírica**. 2018. 74f. Dissertação (Mestrado em Ciência da Computação) – Universidade Estadual de Londrina, Londrina, 2018.

RESUMO

O desenvolvimento de jogos é um processo custoso que normalmente requer uma equipe multidisciplinar. A Geração Procedural de Conteúdo (PCG) pode remediar alguns dos problemas desse processo, auxiliando na criação de diferentes tipos de conteúdos (e.g. fases e gráficos), durante o desenvolvimento e a execução do jogo. No entanto, pouca pesquisa tem sido feita em termos de como a PCG influencia os jogadores, especialmente em Jogos Digitais Matemáticos (DMG). Esta dissertação ataca esse problema investigando como a PCG influencia jogadores de um DMG. Para esse fim, um teste A/B foi executado em que a única diferença foi que uma versão (*estática*) tinha fases criadas por um humano enquanto que a outra (*dinâmica*) provia fases geradas proceduralmente. Para validar esse estudo, um experimento com duas amostras foi designado onde cada amostra jogava em uma única versão. Após isso, dados do jogo e de um questionário advindos de 724 jogadores foram coletados e empiricamente analisados. Baseado nesses, o presente trabalho apresenta quatro contribuições principais: (1) um jogo que encoraja seus jogadores a praticar matemática, desperta a curiosidade e vontade de jogá-lo novamente dos mesmos e que foi avaliado como divertido; (2) mostra que, de acordo com dados do questionário, a experiência dos jogadores da versão dinâmica foi similar à daqueles da versão estática em todos menos um fator, enquanto foi mais difícil e promoveu engajamento equivalente; (3) evidencias empíricas de que a curiosidade dos jogadores tem uma forte correlação com a diversão e vontade de jogá-lo novamente; e (4) evidencias que dados demográficos e de desempenho destes sujeitos no jogo têm impacto de pequeno a moderado em suas experiências. O trabalho discute esses achados, com relação a de onde surgiram essas diferenças, e como eles podem colaborar para o desenvolvimento de jogos, modelagem de jogador e no uso do jogo para fins educacionais.

Palavras-chave: Geração Procedural de Conteúdo. Experiência do Jogador. Teste A/B. Jogo Sério. Jogo Educacional.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

DDA	Dynamic Difficulty Adjustment
DMG	Digital Math Game
GaT	Generate-and-Test
PCG	Procedural Content Generation
PRNG	Pseudo-Random Number Generators
PX	Player Experience
RCT	Random Control Trial
SD	Standard Deviation
U test	Mann-Whitney Test

CONTENTS

1 INTRODUCTION

Mathematics is a relevant subject to citizens' development, impacting on social and cultural aspects, besides academic performance and market labor entering [1]. Some of these arise from its daily use in decision-making, constant global economy's change, and the intrinsic value of its knowledge in culture [2]. Despite that, Brazilian education still faces difficulties in this subject, performing below the expected and between the worst countries according to PISA [3].

This context corroborates with the indication that many students perceive math as difficult, does not like it, and considers the subject displeasing [4]. These might be related to the ease of access to interactive technology of nowadays, which originates students lack interest in the traditional way of teaching [5]. Digital Math Games (DMG) might be used to remedy it, improving students math learning [6, 7] and increasing their positive attitudes towards the subject [8].

Furthermore, DMG can reduce user's anxiety while increasing their engagement [9]. Also, rather than conventional paper and pencil exercise, computer-based practice is preferred by them [10], which can also aid the mentioned problems. Based on this context, the usage of this game's type is fundamental, as can be seen by the attention they have been receiving on the math context [11, 12, 13, 14, 15, 16]. However, even for general purpose games, the development process is still a slow and costly task, which commonly requires several designers, artists, and developers [17, 18].

An alternative that might tackle these problems is Procedural Content Generation (PCG) [19, 17, 20]. It has shown to be a reliable tool, able to provide diversified, automatically generated outputs, which can be controlled through generation parameters [21]. It has been mainly used in games, to automate, aid in creativity, and speed up the creation of various types of game contents [22, 23, 24]. Examples are vegetation, rivers, terrains, networks, scenarios, levels, non-player characters behavior, and control games difficulty level [17]. In addition, it has great potential for educational games [25]; however, it has been scantly used on these games [26, 27, 28].

Moreover, another problem faced in this context is that technologies must provide positive experiences, especially for children. Otherwise, it is unlikely that players will interact or accept it [29, 30]. Developers must be aware that there will be skilled players as well as beginners [31]. Thus, it is necessary to provide them with challenge levels accordingly to prevent them to lose interest [29]. To mitigate this problem, game adaptation might be adopted, as well as constantly providing players with new contents [32, 33, 34].

PCG is a powerful technique to both deliver new contents [22, 21, 35] and to

perform adaptation [36, 37, 38]. For instance, it might be used as a way to change the game's difficulty, increasing or decreasing it according to players performance through the creation of new contents [39, 40]. This approach has demonstrated to benefit the game in comparison to when it is not employed [41]. Especially in educational games, the main benefits it promotes are: motivating effect, more efficient learning, and a way to evaluate students through their gaming performance.

1.1 Problem

Most PCG studies focus on algorithms performance. For instance, the most used evaluation processes are: assessing the generation parameters impact on the algorithm's outputs' expressive range [23]; and modeling players' behavior or opinion in order to personalize or adapt the generated content, and then identify how these models fit the actual players or affect their performance [41, 42].

The influences that PCG usage in the game has on Players Experience (PX)[1] or performance is not captured by those evaluations, though [43]. To accomplish it, assessing the same game with and without the PCG system is the most feasible approach [22]. This approach might be considered an A/B test, wherein two versions - with (A) and without (B) PCG - of a single variable - the same game - are evaluated.

Therefore, this work expands on the literature through the investigation of this gap. Within the DMG context, this dissertation identifies the PCG's influences on players, based on a generic generator, using an A/B test. Hence, both players' opinions regarding their experiences and in-game metrics are used to conduct this research's experiment in a complementary fashion.

1.2 Objectives

The main objective of this dissertation is to identify what are the influences of playing a game with procedurally generated levels in comparison the human-designed levels. Thereby, according to the baseline of a game version that contains human-authored content, we demonstrate *how procedural level generation influences players in a DMG.*

Hereafter, we refer to the game version using human-designed levels as *static* version and the other, which uses procedurally generated levels, as *dynamic*. Also, in the scope of this work, we consider playing a game level as a gameplay, while playing a set of levels is considered a game session. Hence, each level finished by a player (winning or losing it) originates a gameplay. According to these, to achieve the stated goal, the following specific objectives were established:

[1] In the scope of this work, we refer to PX as how their interaction with the game is experienced, following Yannakakis P. Spronck e Andre[42].

- To develop a DMG with human-designed levels to serve as testbed;
- To implement an algorithm to procedurally create levels in a generic fashion;
- To publish this DMG in a way that it captures players feedback and in-game data from one version or another;
- To perform the A/B test considering the collected data of both *static* and *dynamic* versions of the testbed game.

1.3 Research Questions

Four Research Questions (RQ) were defined in order to drive this dissertation's experiments, which are following presented:

- **RQ1:** How does the *static*/*dynamic* game version influence players opinions and performance?
- **RQ2:** Does the *static* and *dynamic* game version differ in terms of how they influence players?
- **RQ3:** Are players experienced fun and willingness to play the game again (*returnance*) correlated to their curiosity?
- **RQ4:** Are players demographics/in-game performance related to their experience?

Through RQ1, we will establish how each game version influence players, showing how both the baseline (*static*) and experimental (*dynamic*) impact them. Thereafter, we will be able to identify whether using *dynamic* version impacts players differently than the *static* (RQ2) and, therefore, capture the PCG's influences. Differently, the subsequent RQ approaches a complementary perspective, extending our main objective.

RQ3 assess if players curiosity is correlated to other PX factors, which is valuable to applications with serious goals once that it is unusual to evaluate curiosity; whereas fun and *returnance* are common. RQ4 concerns how players data are related to their experience. These insights can be used to adapt the game towards improving their experiences, using the game to induce them to be more motivated by its educational content (e.g. curiosity about math) while having more tailored experiences.

1.4 Methodology

Here we describe the process adopted to answer all the previously stated RQ. To this end, we first developed two versions of the same game, to be used as a testbed. They

differ by the level generation processes, wherein one is *static* and the other is *dynamic*. Each one of them is validated with in-game data and players self-reports through RQ1. There exist twenty game levels in the *static* version, therefore, data after playing twenty levels are considered on the analysis of both versions parity.

The data collection procedure and how it was analyzed in order to answer the RQ are described in Section 4. Furthermore, evaluating the same game using PCG and not using it is the most feasible approach is to identify how the employed generator influences players [22, 43, 44]. Thus, we adopt this A/B methodology in order to answer our second RQ and, thereby, identify PCG's influences on players. To accomplish it, data collected to answer RQ1 are compared through empirical analyses.

RQ3 and RQ4 concerns with relationships from players answers between each other and both players attributes and in-game behavior to their opinions, respectively. Hence, empirical analyses are performed to assess correlations/associations between these variables in order to answer these RQ as well. The specific procedure adopted to perform the analysis of RQ one to four is presented in Section 4.3.

1.5 Contributions

According to this work's objectives and RQ investigated, its main contributions can be summarized as follows:

- To introduce a DMG that encourage its players to practice math and provide them with pseudo-infinite game levels and arithmetic problems;
- To show empirical evidence that, besides providing players with positive experiences, this game arises their curiosity;
- To demonstrate that using PCG-created game levels promoted experiences equivalent to human-designed levels in all but one PX factor;
- To reveal demographic characteristics associated with PX as well as how in-game performance is correlated with their experiences;
- To confirm that the difficulty of the *dynamic* game version can be adjusted through the level generation parameter;
- To provide evidence that players experienced fun and willingness to play the game again are correlated to their curiosity;

Therefore, this work contributes to Human-Computer Interaction, Computers & Education, and Game Design, demonstrating: the PCG's influences; presenting and providing a DMG, which from the players perspective, promotes positive experiences and

made them curious; introducing how curiosity is correlated with fun and *returnance*; and showing how demographic data affect PX.

1.6 Outline

This chapter introduced this dissertation's theme and motivated the need for addressing the previously mentioned goal. Also, it showed the RQ that drove this research and the main contributions they provide. The remaining of this document firstly presents fundamental concepts used in its development and a review of similar researches (Chapter 2). Then, it describes the testbed that was used (Chapter 3), followed by the employed experiment design in order to assess the stated objectives (Chapter 4). Thereafter, our experiment's results (Chapter 5) and a discussion about our findings and their limitations (Chapter 6) are presented. Finally, this work's final considerations and proposals for future works are drawn (Chapter 7).

2 PROCEDURAL CONTENT GENERATION

This chapter introduces the fundamental concepts approached in this dissertation. It first describes PCG, what is content in the context of a game, and terms used to specify a PCG system. Then, the search-based PCG approach is presented. Thereafter, the chapter introduces the Experience-Driven PCG, a perspective that creates contents which aim to drive PX. Subsequently, how PCG is evaluated is described, followed by a review of the literature on PCG's impact on PX. Lastly, a summary of how each section relates to this work is presented.

PCG [19] is the algorithmic creation of outputs good enough according to some criterion of the context they will be used [45]. This definition might be clarified with the distinction between *necessary* and *optional* content/output presented in Togelius et al.[35]. The authors argue that both of them are highly dependent on the applied context. Every *necessary* content must be correct, providing the minimal requirements to accomplish the context's goal. In contrast, *optional* are contents that the player might avoid and are allowed to be unusable and/or unreasonable.

For instance, on one hand, in the generation of a maze, the minimal requirement to progress in it, and therefore a *necessary* content, is the existence of at least one path through the initial point to the exit. On the other hand, an *optional* content might be the insertion of enemies or resources that aid the maze's traverse, in which they could be faced/used or not during the gameplay.

As another example, consider the creation of missions (sequence of actions) in a role-playing game. A minimal requirement to players progress might be the possibility to collect the stage's key to complete this mission. If it is available when needed, the content should be considered *necessary*. In contrast, an *optional* content would be the creation of a weapon that does nothing, or hazards that are intended to just difficult the player's path.

The generated contents in those examples, regardless of whether they are *necessary* or *optional*, should be considered good enough. This is true because they are in agreement with their context of application and objective in it. Differently, the generation of a maze's level without a path to the goal point would not be good enough. The presented examples were based on games, which is the main application of PCG [17] and the focus of this dissertation. However, PCG usage is not restricted to it [46, 47, 48, 49].

Furthermore, there are desired properties to achieve common benefits from a PCG algorithm (or generator). Examples are control over its outputs through generation parameters, diverse and expressive results, outputs' structure, and execution speed for some

applications. These properties are listed below:

- **Control** is often desired in many applications with the aim of increasing the content's difficulty or complexity [50], guarantee that these outputs present the designer's intended features [51, 52], and to guide the generation process through parameters without the necessity of understanding the underlying process [53].
- **Diversity** is also important, as one of the main characteristics of using a generator is to have many different outputs. Also, often, it is not interesting that all of them looks like minor variations of the same [23].
- It is desired for a generator to create **expressive** contents, which is able to cover good proportions of metrics space [23]. For example, considering as metric the linearity of a 2D platform level, it is desired that the same generator produce outputs that range from small to large linearity [21].
- For many applications it is expected that the created content features larger **structures** [53] that are internally consistent with its design. It will prevent users from feeling that it is merely a set random noise created by a machine. Although this is interesting for the 2D platform stage, on the generation of more abstract structures (e.g. vegetations), as in the case of SpeedTree [54], this might not be a requirement.
- While **speed** is highly important for applications that use PCG in real time, in a development aided by algorithmically created outputs it is less important. In general, a generator must be able to produce its outputs in an amount of time that satisfies the needs of the application using it [53].

2.1 What is Content?

Content is a key term when PCG is discussed, given that it might be used to refer to multiple elements contained in a game. Examples are: levels [28] (e.g. stages from the Super Mario Bros game); race tracks [55]; missions from a role-playing game, which might be a sequence of objectives to be accomplished [56]; game progression, which might be viewed as the sequence of game levels presented to the player [44]; and music [57].

In Hendrikx et al.[17], the authors surveyed the main classes of game contents that might be procedurally generated for a game. They were structured into six classes: bits, space, systems, scenarios, design, and derived content. The last class does not refer to a content that is strictly inside games, although it is used to immerse players into the game world. Nevertheless, it was considered given that it is derived from the actual game content.

Furthermore, Hendrikx et al.[17] introduced a structure to those classes that enable the identification of which of them may be created with elements from the previous ones. Here, we present these classes following this ordering (e.g. class 2 may be used to build elements from class 3 or 4).

1. **Game bits** are elementary contents wherein the users might interact or that need to be combined with others to allow this interaction. Examples are textures, sound, behavior, vegetations, buildings, fire, water, stone, and clouds.

2. **Game space** refers to the environment where the game happens. It is often the starting point to players construct their interpretation of the game. Mentioned examples are indoor and outdoor maps and bodies of water.

3. **Game systems** might be seen as relationships between game elements. They commonly use complex systems theory and modelings. For instance, they cite ecosystems, road networks, urban environments and entity behavior.

4. **Game scenarios** describe how and in which order the game events will happen. Puzzles, storyboards, story and the concept of a level (playable game space where the player seeks some objectives) are examples.

5. **Game design** are the game's rules, goals, aesthetic and all types of contents, including other game design contents. It can be used to help designers through automatic generation, such as in Mixed-Initiative paradigm [56]. The mentioned examples are the world and system design.

6. **Derived content** is a side-product created by the game world. It is the recorded in-game experiences that can be reviewed and increase the players feeling of immersion. News, broadcasts, and leaderboards are examples.

2.2 Taxonomy

The way that a generator is used, what type of content it produces, and which type of interaction it requires are described here. The presented definitions follow the notions presented in Togelius et al.[35] and Carli et al.[20]. Despite the taxonomy in [35] was originally designed to search-based PCG, it can be adopted on most types of generators.

- Recall from this section's introduction, a *necessary* content must always be correct and is required for the completion of a level. An *optional* content could be avoided or discarded and might be incorrect.

- Contents might be created with a *constructive* algorithm, wherein its results are obtained in a single sequence of step. Thus, it is required to guarantee that its outputs

will be at least good enough during their construction. Differently, a *Generate-and-Test* (GaT) method features two phases: generating and testing, as the name suggests. They are commonly put together in a loop until some generated instance passes the testing criterion, which is dependent on the application context.

- It is possible for a PCG method to be *adaptive* if it considers players behavior and/or profiles to create outputs. This approach is unusual in commercial games, where most of them use a *generic* method that does not take players into account.

- *Online* generation is the case when the content is created in runtime, while the game is executed, as stated in Togelius et al.[35]. It enables the adaptation of outputs and creation of endless gameplay, but requires speed, predictable runtime and, often, predictable quality. In contrast, *offline* generation is when the content's creation is accomplished before the game starts or on the game's development. It can be used to aid designers in creativity or permit the use of methods that are unfeasible in real-time.

- PCG algorithms create their outputs expanding inputs (parameters), which might be a single *seed* or a *set of parameters*. *Seeds* are simpler to use but provides a small degree of control over outputs. On the other side, a vector of parameters allows a higher degree of control, such as specifying the desired features in the algorithm's result (e.g number of enemies and boxes). However, determine the set that yields the desired results is more complex and time-consuming.

- If a generator receives the same set of parameters and creates the same output it is *deterministic*. In contrast, if the method is *stochastic*, this guarantee is inexistent. While one provides reproduce results, the other can be used to achieve diversity.

- An *assisted* technique requires significant human intervention during its setup. Differently, if simple interaction such as just setting up few parameters is needed, the technique is considered *non-assisted*.

2.3 Search-Based Approach

This is a widely used technique in the PCG field, especially for puzzles and levels creation [35]. It enables the designer/developer to guide the generation process towards the intended results. Commonly, some stochastic search/optimization algorithm is used to find a content that satisfies the specified qualities. It is composed of three components: the search algorithm, the content representation, and one or more evaluation functions.

Evolutionary Algorithms (EA) are the most used to form the first component, ranging from simple to sophisticated implementations. The second component is the way in which the generated artifact (content) is represented within the generation process. The

third component is often the hardest task in designing a PCG method using a search-based approach. It is responsible for evaluating each candidate solution and mapping it to a number, that might indicate its quality, playability, difficulty or other evaluation criteria [35].

Content representation is an important aspect of the search-based approach. On one hand, it is a seed that is expanded into the final content, which is the simplest form. On the other hand, it might be an array of numbers, where each number indicates one game's element, and each array index is an output's position. Through these examples is possible to note how the selection of content representation will directly influence the search space and how the final results can be controlled during the search/optimization .

Furthermore, correctly choosing the evaluation function is critical to achieve the expected results on this type of PCG. Each function might be classified into three distinct ways:

- *Direct:* judges the phenotype to determine the content's quality. It might be based on theory, for instance, using designer previous knowledge to determine how the content should be evaluated. Ferreira, Pereira e Toledo[58] applies this approach to level generation using *entropy* and *sparseness* as evaluation metrics. Differently, it might be guided through data. Shaker et al.[59] lies in this class, using a player model created through gameplay and questionnaire data to estimate the content's quality during the generation process.

- *Simulation-based:* guided through artificial agents that play the content to estimate their fitness. This class of evaluation function is used by Luo et al.[60]. The authors present a framework for a data-driven generation, where this process is evaluated by two agents. One is designed to imitate players' behavior and simulate them playing. The other uses this agent to evaluate the generated content effects on real players. While the players' simulation fits as simulation-based, the part where the content's effect is estimated fits as a direct data-driven.

- *Interactive:* involves a human within the estimation process. An example of this class is presented by Cardamone, Loiacono e Lanzi[55]. In their application, players are asked to rate race tracks which are then used in the evolutionary process to create new tracks according to their preferences.

2.4 Experience-Driven Perspective

The simplest form of game adaptation is known as Dynamic Difficulty Adjustment (DDA) [39]. For instance, consider a DDA mechanism in a chess game where a human plays against the CPU. It would decrease the CPU's skills if the player is losing by much,

or increase it otherwise, with the objective of maintaining the player interested in the game.

Difficulty is multi-dimensional, where the same game might be hard for different players due to different reasons. One player could have trouble to solve some puzzles, whereas others could find it easy but consider the necessity of quick reactions hard. However, game adaptation is more than difficulty adjustment and, therefore it should encompass different axes (e.g. fun, challenge, engagement).

Using the experience-driven PCG [40] is one way to achieve this. In summary, it is the use of some computational model able to predict players' behavior/preferences/skills (player modeling), incorporated into a search-based or mixed-initiative approach as the evaluator. It can involve the modification or creation of game contents and produce outputs personalized to specific players.

Thus, it is possible to guide the generator towards the obtainment of outputs that provides the intended experiences to players. These might be challenge, engagement, fun or another modeled experience according to the designers objectives. The main step of the experience-driven PCG perspective is player modeling [42, 61, 62], that later will be used as evaluation function in a search-based approach (see Section 2.3). It is composed by *inputs*, *outputs* and the *modeling approach*, which are following introduced:

- *Inputs* might be of three types: **gameplay** (e.g. scores, playing time, collected items); **objective** (e.g. facial expression, posture, speech); and **game context** (e.g. experienced contents of the game).

- Experience annotation is used to capture model's *outputs*, which might be reported by the player or identified by observation. These annotations can be classified as: **ratings**, being labeled as scalar values (which score you give?); **class-based**, usually a boolean question (was it fun or not?); and **preference**, which compares two or more situations (which one do you prefer?).

- The *modeling approach* maps *inputs* to *outputs*. It might be defined as **model-based** or **model-free**. The former option refers to the assumption of a known mapping from *inputs* to *outputs* in a direct approach. In this case, it might be validated through trial-and-error. The latter considers an unknown mapping that must be discovered, usually by a machine learning algorithm or a statistical model. In this case, there is a dependence of which approach to choose based on the annotation class available. If ratings or class-based are available, algorithms such as Decision Trees and Support Vector Machines might be used. However, if the annotations are in form of ranking, Neuro-Evolutionary preference learning and Linear Discriminant Analysis are the alternatives [40].

2.5 Evaluation

Mainly, there are two perspectives that might be adopted to PCG evaluation. One is focused on the algorithm's capabilities, which is commonly performed through the analysis of the expressive range. In contrast, the other is concerned with how the algorithm's outputs are experienced, which must be captured through players' interaction with it. Following, both of them are briefly presented.

The analysis of an algorithm's expressive range might be summarized in three main steps. At first, the evaluation metrics must be defined, which might be a level's linearity or a difficulty score given by an artificial agent. These will be used to assess a large set of contents (e.g. 10000 levels) generated through the algorithm under evaluation. Finally, this assessment results should be analyzed through plots, such as heat maps and histograms, in order to visualize the generator's expressive range according to the selected metrics [23].

While the aforementioned approach is reliable for investigating how well a method is according to computational metrics, it is insufficient to replace user-based studies [63]. Thus, there is the need of the second perspective of PCG evaluation, investigating how contents are experienced based on studies with real users. Metrics to this end might be questionnaires, observational experiments, facial reactions, voice recordings or physiological responses such as heartbeat intensity [40]. Using these metrics is possible to evaluate the PCG algorithm through its contents based on players' interaction, subjectively and objectively.

Despite that experiments wherein users interact with procedurally generated content capture in which fashion players perceive the contents, they do not provide concerns about the PCG's impact. They show how players perceived the generator outputs, however, how their perception would be if that content was human-authored remains unknown. To actually identify PCG's impact on players, using the same game with and without the generator, in an A/B test fashion, is the most feasible procedure [22].

In sum, PCG might be evaluated through its contents expressive range, focusing on the algorithm's capabilities, through user-based studies, investigating PX according to their interaction with the application using PCG, or through A/B comparisons in order to identify PCG's impact. Hence, the only approach that reveals the influences of PCG usage is the A/B test method.

2.6 Related Works

This dissertation concerns how procedural level generation influences players of a DMG in comparison to human-designed content. Thereby, we survey related works which

perform similar experiments, executing A/B comparison on the same game, comparing procedurally generated content to human-authored content according to their experiences.

Korn et al.[22] evaluated the use of a procedural generation system based on players' self-reports through a documentary game. Their generation system was in charge of creating game's reefs, elements that were compared to human-designed ones according to a total of 41 subjects. These subjects were adults, which played both game's versions (10-15 minutes each) and responded to a questionnaire after playing each one. Players indicated their experiences based on reefs visual aspects and preference for one version or another, favoring the automatically generated contents in both perspectives. Also, they found that older players were most likely to favor the procedurally generated reefs. Their findings show that PCG can provide games with more than money saving, influencing PX positively, and that game environment's change is an advantage.

Connor, Greig e Kruse[43] performed a similar research, investigating PCG impact's on players' immersion using an abstract game. However, they evaluated the generation of levels instead of only creating an element from it. Twenty players participated in this research, where they were separated into two groups, playing the game version with human-designed levels or the one with levels created through PCG. Both samples were also composed of adults, between 18 and 35 years, who responded to a player immersion questionnaire (30 questions) after playing one of the two versions only.

Unlike the results of Korn et al.[22], Connor, Greig e Kruse[43] found a significant difference between these versions in favor of the human-designed contents, considering their total immersion, which might have arisen due to the generation of levels rather than a single element type. However, when analyzing each questionnaire answer, the authors found that this difference was significant only for less than 17% (5/30) of the questions. The small sample size is a limitation which might have affected these results, as well as the fact that some parts of their level (e.g. level boundaries) were not created through the PCG algorithm. Thus, evaluating an approach where the entire level is algorithmically designed, based on a larger sample, could mitigate these drawbacks [43].

Despite that Butler et al.[44] addressed a research methodology similar to the aforementioned studies, an educational math game was adopted as the testbed, unlike the previous ones. Also, rather than performing analysis based on players' self-reports, in-game engagement metrics were used. As the adopted game was online and collected data from an uncontrolled environment, players demographics were not available due to the nature of the environment wherein their testbed was hosted. A two-sample analysis was also used in this research, comparing both the time and amount of levels played based on data from 2377 players. For this research, similar to Connor, Greig e Kruse[43], the PCG system creates the game levels, however, focused on creating a progression based on the solution of the math problems (fractions).

Alike the results of Connor, Greig e Kruse[43], Butler et al.[44] found a significant difference between game versions, in the amount of played levels specifically. Also, they showed that this difference was small, which was not report on previous surveyed papers. With respect to the total time that each version was played, the difference was insignificant although the samples size. Even though, the PCG based one was played approximately 92% of the time that the human-designed, showing that their solution is capable of engaging players for similar amounts of time in comparison to human-authored.

2.7 Summary

In this work, level generation in an adventure game will be investigated. It fits in content's class number four (scenarios) according to the definitions presented in Hendrikx et al.[17]. They argue that this is one of the most popular types of PCG for games and that nearly all genres can benefit from its usage. Also, the testbed game used in this work will benefit from PCG to create its math challenges. According to the same definitions, these challenges might fit in class number four as well, since it can be considered a puzzle.

Both of these contents' types are key elements to the game's objectives, which the players do not have the option to avoid them in order to advance in the game. Thus, the two must be considered *necessary*. For the sake of comparison, both game versions will provide endless gameplays. Then, in the specific case of using human-designed levels (*static* version), once the player wins the last level available, it starts to play the other version.

A *constructive* method will be used to create the math problems and the *generic* level generator. This method is *generic* since it uses the same approach for all players, although contents difficulty increase as players' wins sequence increases. After these generators are developed, they just require parameters setup due to their *constructive* nature. Therefore, they are considered as *non-assisted* techniques. Nevertheless, both level and math problem generation methods are *stochastic* and each one receive as input a *set of parameters*, as discussed in Chapter 3. Lastly, procedurally generated levels have their influences evaluated, which was performed on most studies that also guided our experiment's choices, as described in Chapter 4.

3 TESTBED

The purpose of this work is to demonstrate how procedural level generation influences players of a DMG. Fundamentally, it is focused on identifying if players have their experiences and behavior within the game influenced. Performing an A/B comparison is the most feasible approach to this end [22], which led us to develop a DMG that meets this needs to serve as a testbed. Thereby, this game features human-designed levels as well as a PCG system that creates levels based on a *generic* approach.

Considering this context, the remaining of this chapter introduces the resources used to implement this testbed. Firstly, it introduces the developed DMG, *SpaceMath*. Then, the chapter presents the game's main features, namely the arithmetic problems generation method, the human-authored levels and the generic PCG algorithm for levels creation.

3.1 *SpaceMath*

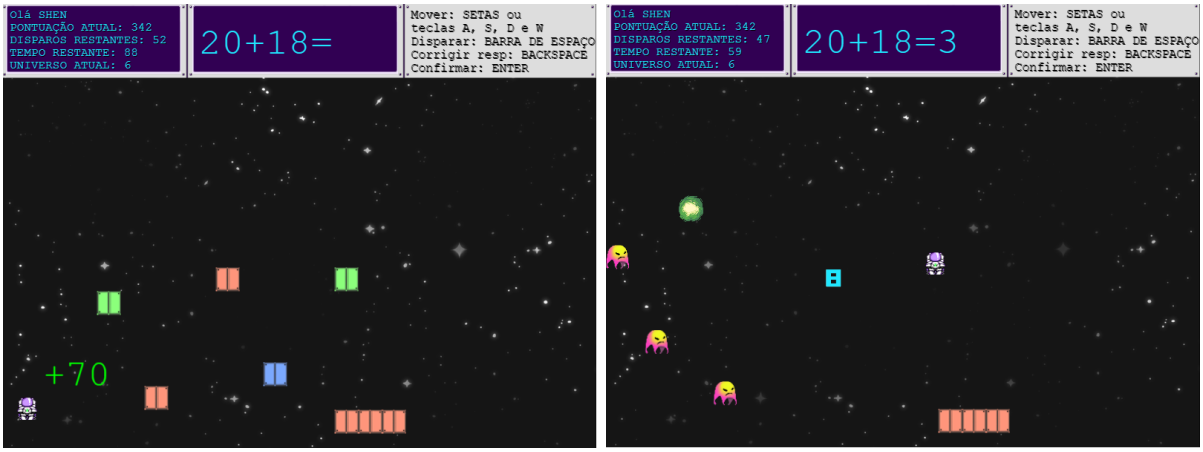
Inside *SpaceMath* [64], the player incorporates an astronaut. It is lost in a parallel dimension on space and must solve arithmetic puzzles to escape from it. The game aims to make players solve as many puzzles as possible. Thus, even after a win, the player is transported to another parallel dimension, rather than actually escaping.

It is an adventure game that encourages players to solve math challenges, more specifically, the four basic arithmetic operations. These are key concepts for math, taught at the beginning of students' academic journey and relevant to many subjects (e.g. algebra, calculus, and physics). Thus, this game might be valuable to children, teenagers, and adults, aiding them to train and/or strengthen their arithmetic background in a lucid fashion.

Players' objective is to find numbers and collect them, in an order such that their concatenation forms the challenge's answer (e.g. collect 3 and 8 if the answer is 38). They are allowed to drop back a number if they judge it necessary to formulate the correct answer. These numbers are hidden by boxes, in which the astronaut must shot to teleport them out of the current level. After that, it might reveal one of the answer's numbers or spawn zero, one or two aliens.

Figure 1 presents a randomly selected level as it was generated and after the player explored it for 31 seconds. Time does not change levels, it aims at making players quickly solve the challenge, preventing them to use other devices to do it, and to limit their time to explore the level. Additionally, the figure shows the astronaut and the score's feedback received after winning the previous level (i.e. +70) at the bottom left of Figure 1a. Also, it

shows the level's boxes (might be green, tangerine or blue), aliens (Figure 1b bottom left) and portal (Figure 1b middle left). The number eight, that is one of the answer's numbers (Figure 1b middle), is displayed as well. At the figure's top, player's status, level's problem and current answer (three), and playing keys are displayed from left to right (all gaming info are in Portuguese).



(a) Initial arrangement. (b) After exploration.

Figure 1 – Interface randomly selected of *SpaceMath*.

To confirm its answer, the player must go inside the portal, which appears after some number (three in the example of Figure 1) is collected. If the numbers were collected in the expected order, formulating the right answer (i.e. 38 on Figure 1), the player advances to the next level (a new parallel universe), receives a charge on his device (plus 10) that is 10 at the first level, and has its score increased by 30 plus half of the time left.

In contrast, if the player misses the answer, is touched by an alien or runs out of time (90 seconds): it is declared a loss, the game restarts from the first level, and player's score is reset to zero. Nevertheless, players' scores are always stored at the database and used to build up a leaderboard. It displays the 15 highest scores, considering the high-score from each player to avoid its appearance twice. The goal is to motivate players to compete with each other and continue to play.

Level's boxes might be grouped into three categories: numbered, which hides an answer's number; hazard, that hides nothing, difficult players' path, and might lead them to waste device's charge in it; and harmful, which spawns one or two aliens. Aliens and boxes might be teleported to another dimension through the astronaut's device. This will use the portal's energy to recharge the device (plus three).

Furthermore, boxes have their visual presentation modified as the game progresses. As can be seen in Figure 1, they might have three different colors, which are used to guide the player towards the correct answer. Thus, it is possible to identify that a green box is hiding the dozen and that the unit is behind a tangerine box when the answer has two

digits.

This is illustrated by Figure 1, where the numbers three and eight were under green and tangerine boxes, respectively, and the problem's answer (38) has two digits. Note that colors indicate collecting ordering, not fixed units, dozens or hundreds. If the answer was greater than 99, let's say 110, a box of the third color (blue) will appear in the level. In this case, the player would be required to collect the numbers hidden under the green, tangerine and blue boxes, in this exact order.

However, this approach is used only on the first 10 levels, whereas the subsequent levels contain boxes with a single color (white). In spite of that, it is left to the player to identify this guidance, as well as manage its time and device's charge left. Moreover, another particularity of *SpaceMath* is that it contains two playing versions: *static* and *dynamic*. The former contains levels created by a game developer, whilst the dynamic uses a PCG to create them. Nevertheless, the game's system is in charge of managing which version will be playing, not the player itself.

3.2 Arithmetic Problems Generation

This generation process is *generic*, the same for all players, and happens *online*, previously to each gameplay starts. We highlight that this generation's influences are not investigated in this work. Despite that, it is presented here for the sake of completeness as this is a key mechanic of the testbed game.

Every game challenge has exactly two numbers (e.g $10 + 2$) and it is one of the four basic arithmetic operations (summation, subtraction, multiplication, and division). At each level, a Pseudo Random Number Generator (PRNG) chooses one operation of the available list (initial arrangement = [+, -, *, /]) and then remove it. When this list becomes empty, it is reconstructed with the initial disposition.

Each problem's number respect a specific parameter, which is updated as the game progresses. These parameters are used as upper bounds on the PRNG to determine the problem's values (default lower bound is zero). Initially, both parameters are set to five and the update value, which is the same for both, is set to 10. At odd levels, only the first value is updated, and the second only on even ones. The maximum value for them is 100. When a player loses, both parameters are reset to the initial values. The problem's generation process is illustrated in Algorithm 1.

The parameters *op* and *MV* are chosen as described above and the PRNG is instantiated at the game's start. As can be seen in Algorithm 1, this procedure has four main steps: (1) initialize the output's object; (2) choose the problem's numbers; (3) sort them; and (4) guarantee that results are integer numbers in case of divisions.

Algorithm 1: Arithmetic Problem's Generation

Input : *op*: operation type
MV: set of two numeric values indicating the PRNG's upper bounds
Output: *mp*: arithmetic problem object with its set of numbers and operation

```
1  begin
2      mp ← empty object
3      mp.nums ← ∅
4      mp.op ← op
5      for i ← 0 to 2 do
6          minV ← 1
7          if op is multiplication then
8              minV ← 2
9              MV_i ← max(5, ⌈MV_i ÷ 5⌉)
10         end if
11         rn ← PRNG.randomInteger(minV, MV[i])
12         if op is division then
13             mi ← round(|NPL| × MV[i] ÷ 100)
14             rn ← NPL[PRNG.randomInteger(1, mi)]
15         end if
16         Append rn to mp.nums
17     end for
18     Sort mp.nums in decreasing order
19     if op is division then
20         while mp.nums[0] is not divisible by mp.nums[1] or
21         mp.nums[0] = mp.nums[1] do
22             mp.nums[1] ← PRNG.randomInteger(2, ⌊mp.nums[0] ÷ 2⌋)
23         end while
24     end if
25 end
```

The second step contains some exceptions, regarding the generation of multiplication and division problems. As previously stated, this is a *constructive* algorithm. Thus, it is necessary to solve any possible problems during the output's construction to prevent unreasonable contents. There are two exceptions in multiplications: each parameter is adapted in order to avoid results too large (line 9); and the PRNG's lower bound is set to two (line 8), avoiding multiplications by one.

Divisions have another two exceptions. One is that prime numbers are excluded from selection, using values drawn from a *Non-Primes* List (*NPL*). The *NPL* contains all non-prime numbers from 1 to 100 in increasing order. In this case, the PRNG chooses a list's index, ranging from 1 to *mi* (line 13). This index represents its proportion to be considered on the number selection. The second division's exception is that fractional results are guaranteed to be inexistent (lines 18-24), which is optimized through the *NPL*

list's use. In addition, by sorting the problem's numbers, positive results in all subtractions are guaranteed.

3.3 Human Authored Levels

This version contains *static* levels (total of 20) that were designed by a game developer. He is graduated in Technology in Digital Games and had three years of experience in the field at the time of designing them. While the game is being played, the order in which these levels are presented increases the number of elements in them according to the sequence of wins that the player currently has, with the aim of raising their difficulty.

In this game's version, in the case of a loss after advancing three levels, the player will restart from the beginning and have to play on the same three levels again. However, in the case of reaching a 20 wins streak, the player is redirected to the *dynamic* version. Also, after achieving this point, the player will play in the procedurally generated levels from the next login. Despite that, due to the questionnaire being presented after playing 20 levels, their answer is guaranteed to be based on playing the *static* version only.

The level's difficulty progression is based on the following specifications. The first levels have few boxes and no aliens are present beneath them. Then, after each win, the amount of boxes is increased and harmful elements progressively begin to appear. Thereafter, boxes of the third color (required only if the answer is bigger than 99) are inserted, seeking to lure players to waste the device's charge and difficult their path. Also, the amount of aliens is progressively increased.

Note that the arithmetic challenge is automatically generated in both *static* and *dynamic* versions. Therefore, it might be necessary to adapt some boxes color or hide a number in a box which was initially designed to be a hazard. For example, consider that a challenge with answer greater than 9 is created in a *static* level initially designed with a single numbered box. In this case, the game will search for an existent tangerine hazard (color of the second piece of the answer, as mentioned before) and hide the number in it.

There is also the case when the correct color does not exist. In the aforementioned example, a numbered tangerine box. A similar case would be if the answer was > 99 and no blue color was originally in the level. For these cases, the game will randomly find a hazard, change its color to the correct one and hide the number there.

3.4 Generic Level Generation

Similar to the problem's generation, the constructive algorithm [65] of *generic* levels generation also runs *online*. Figure 2 presents a high level representation of this process. Firstly, the level is discretized in order to represent it as a grid (1). Then, the

numbered elements (boxes or blocks) are positioned (2). Thereafter, traps (3) and hazards (4) are created. The last step (5) is to, optionally, repeat steps three and four, which is stipulated by the generation parameter w. Following, complementary intuitions about this procedure are provided.

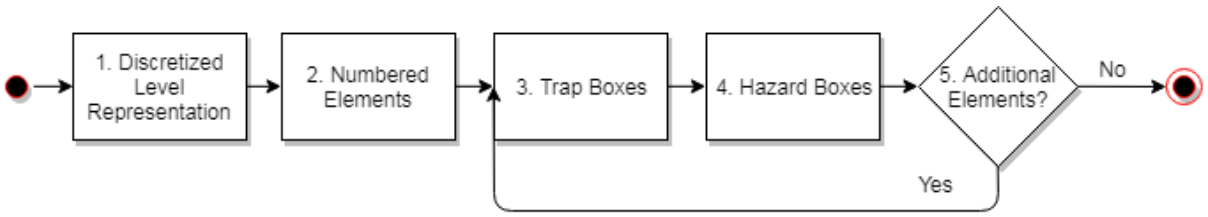


Figure 2 – High level representation of the generic level generation procedure.

One of our goals with the level generation process was to use it to represent, within the game, the arithmetic problem resolution. Thus, we used the size of the problem's answer (a) as one of the generation parameters. The other parameter is the player's wins streak (w), which allows the control of the levels' difficulty. This procedure is further represented in Algorithm 2.

As can be viewed in Algorithm 2, the constructive generation begins by initializing an empty level representation, which discretizes level's coordinates to represent it as a grid (lines 2-12). This discretization is accomplished according to sprites size (ss), that is the same for them all. Thereafter, the player's avatar is spawned, always on the scenario's bottom left (line 13). The avatar's position and its neighbors are not available as empty positions from the beginning (line 6).

Using the PRNG, a split index (si) between 0 and a is generated (line 14). It will define how many numbers will be below single boxes (a - si) and below blocks of boxes (si). The order of boxes and blocks creation is also based on a PRNG choice (lines 15-22). This is important on levels where boxes give tips. In this case, a repeated sequence would often imply on units being in a single created box and dozens in blocks, or vice versa.

Specifically, a numbered block is a rectangular cluster of boxes which contain one answer's number. Its size indicates both its number of rows and columns. For instance, a block with size equals three is expected to have three rows and columns, with a total of nine boxes. However, if any of these boxes would be created out of the level's bounds (vertically or horizontally), or over an existent element, its creation is ignored. Thus, it might be smaller than the expected.

Each numbered block's size is specified by the PRNG: an integer between 2 and $\max(3, \lceil \log_{10} w \rceil * 2)$. The max and logarithm functions prevent blocks too small or too large, respectively. Also, the PRNG decides which block's box will hide the number, randomly selecting one of them. In case it chooses an unavailable level position (i.e occupied by a previously generated element), it will choose the last created box to be numbered.

Algorithm 2: Constructive Level Generation

Input : a: size of the answer of the arithmetic's problem
w: current player's win streak
os: harmful elements offset
ss: sprites size

```
1  begin
2      l ← level representation
3      positions ← ∅
4      for i ← 0 to l.width ÷ ss do
5          for j ← 0 to l.height ÷ ss do
6              if i > 3 and j < l.height − 3 then
7                  pos.x ← i × l.spriteSize
8                  pos.y ← j × l.spriteSize
9                  append pos to positions
10             end if
11         end for
12     end for
13     l.avatar.position ← bottom left
14     si ← PRNG.randomInteger(0, a)
15     p ← min(10, ((w + 1) − os) ∗ 0.5)
16     if PRNG.randomInteger(0, 2) = 0 then
17         create a − si numbered boxes at random coordinates from positions
18         create si numbered blocks at random coordinates from positions with
            p chance to hide an alien in each box
19     end if
20     else
21         create si numbered blocks at random coordinates from positions with
            p chance to hide an alien in each box
22         create a − si numbered boxes at random coordinates from positions
23     end if
24     create w − os aliens at random coordinates from positions
25     create a harmful block at a random coordinate from positions with p
        chance to hide an alien in each box
26     for i ← 1 to (w - os) do
27         if PRNG.randomInteger(0, 100) < min(50, (w + 1) ∗ 5) then
28             create a harmful block at a random coordinate from positions
                with p chance to hide an alien in each box
29         end if
30     end for
31 end
```

Both numbered elements generation are accomplished using the PRNG to choose empty coordinates from *positions*. A particularity of this generation is that, on one hand, a numbered box will never spawn an enemy. On the other hand, a numbered block has a p chance of hiding an enemy, in each one of its boxes, except in the box which actually hides the number. The variable p is defined in line 15, where $w+1$ represents the player's

current level and *os* is the harmful offset of early levels. Also, it bounds this probability to range from 0-10 probability.

The offset ($os = 3$) was defined to avoid harmful elements on the first three levels. Therefore, $w - os$ harmful boxes and at least one block (lines 24-25) are created, with size between (2, min(4, w + 2)) chosen by the PRNG. However, $w - os$ additional blocks, with size defined in the same way, might be generated with a probability between 0.05 and 0.5 (line 27), for each one. In comparison to numbered blocks, the difference is that these do no hide answer's numbers. However, they have the same probability of hiding an alien.

Every time a harmful box is destroyed, at least one alien is spawned in its position. In addition, with a $(w + 1) * 0.1$ probability, an additional enemy is spawned in a random coordinate from *positions*. Note that the exit portal creation is not in Algorithm 2, which is executed previously the gameplay starts. Differently, the portal is spawned after the player collects any piece of the answer, being positioned at a random coordinate from *positions*.

4 EXPERIMENT DESIGN

This chapter introduces the design of the experiment that was employed in this work. At first, we introduce how its execution was planned, which might be summarized as follows:

1. Publishing the testbed game
2. Collecting data
3. Analyzing samples demographic data
4. Defining self-reports and in-game data analysis process
5. Empirically analyzing self-reports and in-game data

After developing the testbed game, the first step was to publish it online, making the game available for free access of anyone through a computer with internet access. Then, we sought for volunteers through email lists, social networks disclosure, and colleagues. Thereafter, we investigated our samples demographic characteristics, followed by the analysis of data regarding their interactions with the game as well as their opinions.

In the remaining of this chapter, steps two to four are presented. Firstly, we describe how data collection was executed, showing which data was gathered and how it was captured. Then, we perform the analysis of both Control and Experimental samples' demographic characteristics. Lastly, we describe the process that was executed to answer each RQ, presenting metrics and procedures performed. Step five's results are shown in Chapter 5.

4.1 Data Collection

Despite the game is available online and constantly captures data, this experiment was performed in four institutions (three private and one public), always under the supervision of a researcher and an institution's supervisor (teacher or professor). While these its execution in these institutions was possible due to colleague teachers the performed it, some players were reached through the internet. Hence, over 70% of our data were gathered on supervised applications, whereas the rest was collected in the wild, outside researchers or supervisors control.

Before starting to play the game, players were introduced to the testbed game and, then, had to create an account. Besides an identifier nickname and a password, they

had to answer other nine questions. These data are their demographic features, which are shown in Table 1 along to a brief explanation of how they were encoded.

Table 1 – Players' demographic features.

Feature	Type	Description
Age	Continuous	Current user age
Genre	Boolean	Whether the player is male (1) or not (0)
HasNet	Boolean	Has a computer with internet access at home (1)
SchoolType	Categorical	municipal (0), public (1), federal (2) or private (3)
SchoolYear	Categorical	Between 1^{st} and 9^{th} year, or finished (0)
PlayingTime	Continuous	Average gaming time per week in hours
Gamer	Boolean	Considers itself a gamer (1) or not (0)
LikesMath	Categorical	How much enjoys math in a five-point scale
KnowsMath	Categorical	Knowledge in math in a five-point scale

As can be seen on Table 1, basic personal information (age and genre), academic data (school type and year) and internet access at home through a computer (HasNet) were collected. In addition, playing habits (gamer and playing time) and personal opinion about math (likes and knows math) were asked as well. Regarding the likes math statement, it ranged from *I do not even want to hear about it* (1) to *Yes, I consider it the most important subject of school* (5). The knows math ranged from *very low* (1) to *very high* (5).

After the game's initial period online, we noticed that *schoolyear* had too restrictive answer options. Some players had finished elementary school already, providing little information. Thereafter, we replaced it by a question which captures in which school's stage players are (e.g. elementary, middle, etc.). Subsequently, we merged these attributes wherein players with register indicating finished elementary school where categorized as Others/Unknown, while new players could choose between the Brazilian's school stages.

Furthermore, during the gameplay, in-game metrics were constantly stored after each played level. Using them, we created a performance history of each player, which is composed by the average of some of these metrics. To a better description, Table 2 presents the features composing it. Based on the table, it is possible to notice that, for each level, metrics such as score, time spent, shots and sequence of wins were stored. These metrics enabled the creation of all the features presented in the aforementioned table.

Moreover, players were asked to answer a PX questionnaire after playing 20 game levels. It captured four factors: experienced fun; *returnance*; curiosity; and experience description. Its composition was based on previous studies which already used and validated them. Two of its parts are based on the widely used Fun-Toolkit [66] and the others are inspired by its use for rapid assessment [67]. Hence, our questionnaire not only evaluates

Table 2 – Performance history features (all continuous).

Feature	Description
AvgScore	Average score per level
MaxScore	Maximum summed score achieved
AvgShots	Average of shots fired per level
AvgTime	Average of time spent to complete each level in seconds
SumTime	Total time spent playing the game in seconds
MaxSeq	Largest sequence of wins achieved
TotalPlayed	Total of played levels
SumWins	Total of wins in all levels

enjoyment (fun) and re-use (*returnance*) but curiosity, which is of utmost value to educational systems, and description of the experience, that is related to enjoyment, however, assessing it in a deeper way. Following, each one of these factors is explained and how they were captured by the questionnaire is described.

Experienced Fun was captured using the Smileyometer from the Fun-Toolkit [66]. It provides a simple and intuitive way to players indicate this factor. It was encoded as a rating, on a five-point scale ranging from 1 to 5, where higher values indicate more fun.

Description of Experience investigates PX in a deeper way than the Smileyometer. It is based on predefined opposed attributes in order to have a semantic balance. This approach was inspired by its usage in [67]. The following attributes were captured in a class-based way: simple - difficult; great - childish; fun - boring; exciting - tiring; and intuitive - confusing.

Returnance identifies players' willingness to play the game again (class-based). In other words, it asks users to indicate if they would play the game again, choosing between yes (5), maybe (3) or no (1). This questionnaire's section was based on another tool from the Fun-Toolkit, the Again Again Table [66].

Curiosity was adapted from the questionnaire used in [68]. It was captured through the following questions, that were also encoded as ratings in a five-point scale ranging from 1 (completely disagree) to 5 (completely agree):

- The game motivated me to learn more about math;
- I wanted to continue playing because I wanted to see more about the game levels;
- Playing the game raised questions about the game levels;
- I was curious about the next event in the game;
- I sought explanations for what I encountered in the game;

- Playing the game raised questions regarding math;
- I wanted to continue playing because I wanted to know more about math.

Thus, our PX questionnaire has four sections and a total of 10 questions. In nine of them the players must choose a single option and in one (experience description) they can select up to 10 attributes. We highlight that our target sample are Brazilian players and, therefore, every question was translated to Portuguese. In sum, these three types of data will allow us to conduct a more robust research about the presented proposal influences.

4.2 Samples

The experiments reported in this document are based on a sample of 724 players registered in the game. Thereby, we managed to identify which players belonged to which group (Control or Experimental). They were distributed between the two groups using RCT at the time they registered in the game. In this document, Experimental (N=369) refers to players who interacted with the *dynamic* game version. In contrast, Control group (N=355) is composed by the ones who played in the *static* version.

RCT was achieved using the numeric identifier from each player's register in the database, which is determined automatically (auto-increment). Thus, it is possible to provide them with the same game version as before on new game sessions, based on their identifier. Hence, players were distributed between groups without any human bias, wherein an even or odd identifier led players Experimental or Control group, respectively.

Due to the two-samples research design, it is necessary to investigate whether these populations differ from each other. This is important to avoid biases from players with different characteristics since a key goal of this work is to compare their answers. At the first step, their demographics' distribution were assessed, which provided significant evidence that these attributes do not follow a normal distribution. Therefore, we examined groups' differences through non-parametric hypothesis tests based on players that responded the questionnaire.

From the demographics characteristics captured on the registration questionnaire, two of them were numeric: age and weekly playing hours. Thus, we compared these attributes of both groups through the Mann-Whitney U test (U test, hereafter). Figure 3 demonstrates age distribution of players from both groups. It shows that the majority of players were between 10 and 17 years and that the age with most players (11 years) had less than 30% of the full sample. Also, it shows that players age were distributed alike, which is confirmed through the U test ($U = 31860$, $p = 0.9005$), where the average age of control is 14.1 years ($SD = 5.7$) and experimental's average age is 14.5 ($SD = 6.5$). Thereby, the sample's had an insignificant difference in terms of age.

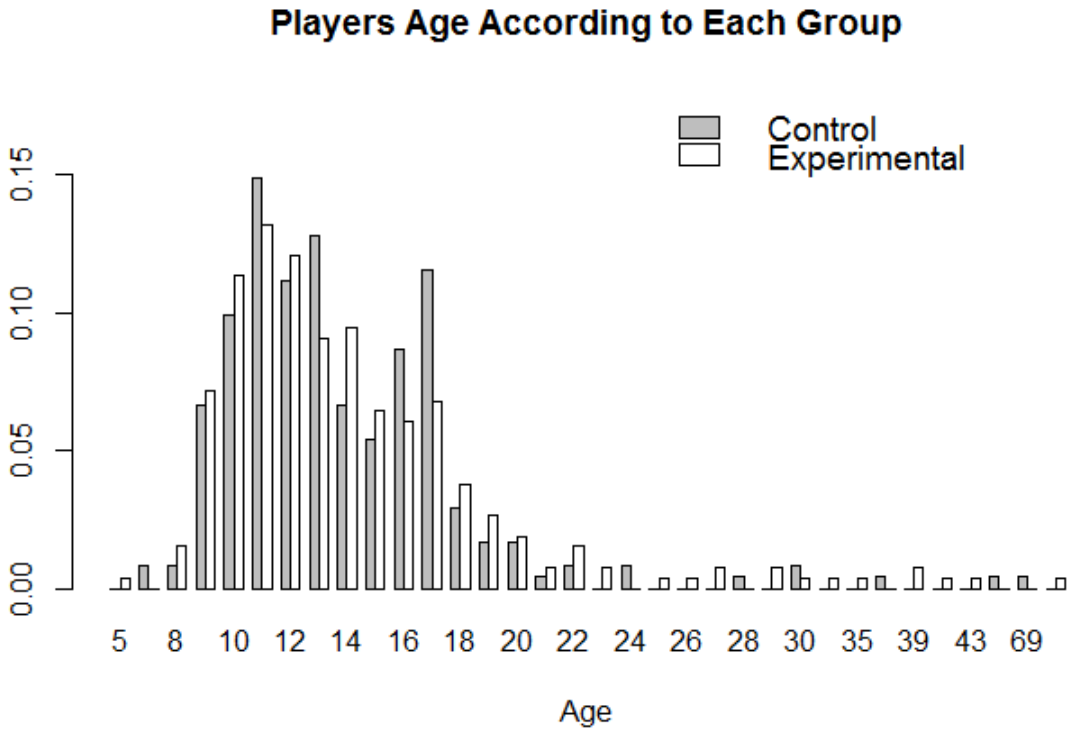


Figure 3 – Barplot of full sample's age distribution.

Concerning the other numeric demographic characteristic, the time players play during a week, we wanted to know whether players from distinct samples weekly play different amounts of time. It was assessed through the U test as well, showing that this difference is also insignificant between groups (U = 32943, p = 0.5931) and that both of them approximates the average time Brazilians play, 15 hours per week [69]. The average playing time from players of the control group is 14.1 hours (SD = 25.6), whereas the average of experimental's group is 13.3 hours (SD = 26.4). Thus, in terms of hours played during a week, there is not a significant difference between the analyzed samples and they approximate the time of the actual population.

In relation to the remaining demographics, which are categorical, Table 3 presents their distributions across classes. With respect to males, gamers and the ones with internet access through a computer at home, the table shows the number of players that were/had (Y) and were/had not (N). Regarding the school type, classes were abbreviated as M (municipal), E (public), F (federal) and P (private). Finally, math-related characteristics were abbreviated according to the five-point Likert scale, ranging from completely disagree (CD) to completely agree (CA). Also, the table presents the results of the χ^2 homogeneity test, which evaluated whether the answers' distributions were the same for both samples according to the five attributes. Significance would be denoted by an asterisk, however, there were no significant differences between groups, as can be seen in the table.

Table 3 – Comparison of samples categorical attributes.

Attribute	**Control**	**Experimental**	χ^2 **(df)**
Males	N=79, Y=163	N=106, Y=159	2.644 (1)
Gamers	N=122, Y=120	N=139, Y=126	0.137 (1)
Has Net	N=27, Y=215	N=23, Y=242	0.617 (1)
School type	M=6, E=72, F=4, P=160	M=5, E=61, F=7, P=192	3.692 (3)
Likes Math	1=30, 2=52, 3=31, 4=82, 5=47	1=20, 2=46, 3=41, 4=90, 5=68	6.934 (4)
Knows Math	1=15, 2=24, 3=140, 4=50, 5=13	1=8, 2=24, 3=149, 4=65, 5=19	4.458 (4)

Thereby, we can conclude that there is no significant difference between the samples under analysis in terms of the eight attributes evaluated. Hence, it is expected that subjects with distinct characteristic will not bias this research's results.

4.3 Data Analysis Process

RQ1 is concerned with how each game version influence players in two perspectives. One is regarding their experience, which is captured through self-reports, while the other is focused on their performance, which is captured through in-game metrics. R project [70] was used to accomplish all statistical tests performed in this work. Effect size was calculated using the *rcompanion* package through the *wilcoxonR* function since the initial analysis provided evidence that these data do not follow a normal distribution. The usual alpha of 0.05 was adopted for all hypothesis tests.

Empirical analyses were performed to identify how game versions influence players. In terms of their opinions, for experienced fun, *returnance* and curiosity, one-sample hypothesis tests were used to investigate the hypotheses that players agreed (> 3) with the questionnaire statements. The one-sample U test [71] was be used to this task. In terms of playing behavior, two perspectives were evaluated. One is referred to as Retainment, which are players who answered the PX questionnaire (played 20 levels or more). The second is referred to as Engagement, which considers how many levels were played on average, similar to Butler et al.[44]. For RQ1, these aspects were evaluated through the average of players who were retained and the average of levels played by each one of them. Lastly, regarding in-game performance, the goal is to analyze it concerning how it was at the point of answering the questionnaire. Hence, performance is assessed considering in-game data until the point that players provided their self-reports.

RQ2 concerns whether the influences of the *static* game version differs from the ones of the *dynamic*, also considering both their self-reports and in-game behavior, as well as their performance. To do so, we followed the literature and used the same ap-

proach than Connor, Greig e Kruse[43] and Butler et al.[44], which performed similar evaluations. Therefore, players opinions were compared through the Kruskal-Wallis test [72], while numeric behavioral data are compared through the U test. Retainment was assessed through the Chi-Squared (χ^2) homogeneity test [73], investigating whether the distribution of retained and non-retained players are the same for both Control and Experimental group. Considering the amount of played levels, we will also use the U test to check if the average of played levels is the same both groups.

When assessing the experience a game promoted, researchers usually focus on experienced fun (e.g. [74, 75]) and *returnance* - or retention - (e.g. [76, 77, 78]). Differently, this work also investigates players curiosity, which is valuable to serious games, capturing it through a set of seven questions here. Thus, we want to know how these factors are correlated to each other (RQ3), accomplishing it through Kendall's rank correlation test [79]. Mainly, it was selected over Pearson's test due to data's measure, which is neither interval nor ratio, and over Spearman's due to the number of tied ranks [80, 81].

Nevertheless, for the sake of an easier interpretation, Kendall's correlation coefficients (τ) are transformed into Pearson's (r) [82] and presented as well. The degrees of correlation are interpreted as large/strong if greater than 0.5, moderate if between 0.3-0.5, small if 0.1-0.3 and insubstantial/trivial otherwise [83]. Despite that, correlation significance is based on Kendall's test. Therefore, we analyze a correlation matrix, encompassing all of them to find if they are significantly interrelated and the degree of these interactions.

Finally, in order to find whether demographic characteristics and in-game performance are related to PX, and thus answering RQ4, two tests were used. One is Kendall's rank correlation test [79], that will be employed to assess continuous and ordinal characteristics (e.g. age and likes math), similar to in RQ4. The second is the χ^2 independence test [73], which is applied to dichotomous attributes which cannot be ranked (e.g. genre). A similar procedure was performed in Korn et al.[22] to investigate how the players' reports are associated with their age.

5 RESULTS

This chapter presents the results of the experiment reported in this document. Thereafter, a summary of our findings is presented in terms of how they answer each RQ of this study.

5.1 Static Game Version Influences

This section partially answers RQ1. At first, how this game version influenced players opinions is presented. Then, the performance and behavior of players from the control group are examined.

The PX questionnaire was answered by 242 players from this version. Table 4 demonstrates the PX factors statistics (average, median and standard deviation), the Mu value used in the test to compare the answers' median, the test statistic (U), and the effect size for fun, *returnance* and curiosity as a whole. As can be seen in the table, the three factors had a large highly significant difference from the selected Mu, which represents the indifferent answer. Thus, we can conclude that the static version had a positive influence on players, promoting fun, willingness to play the game again and curiosity.

Table 4 – Control group's self-reported experience.

PX Factor	**Mean**	**Median**	**SD**	**Mu**	**U**	**Effect Size**
Fun	4.446	5	0.778	3	23897.500*	0.855
Returnance	4.397	5	1.039	3	16192*	0.803
Curiosity	3.902	4	1.008	3	853812.500*	0.668

* $p < 0.01$

Table 5 concerns these players from another perspective, showing their in-game performance until the point of answering the questionnaire. It demonstrates the summary (minimum, first quartile, mean, median, third quartile and maximum) of seven in-game metrics. As can be seen, there were varied performances, wherein the minimum win rate was 60% whereas the maximum was 100%, for example. Also, it can be noted that while the average player achieved the maximum score of 536.4, the best one achieved almost 1200. Thereby, players who interacted with this game's version and responded to the PX questionnaire performed in distinct ways, yielding high and low results according to the evaluated metrics.

In terms of in-game behavior, considering the 355 players, an average of 53 levels (SD = 53) was played by them. From these, 96 did not answer the questionnaire and played less than 20 levels. Hence, 73% of them were retained by the testbed game according to

Table 5 – Summary of control group's in-game performance.

Metric	Min.	1st Q.	Median	Mean	3rd Q.	Max
Avg Score	34.15	51.66	55.69	54.74	58.50	65.58
Avg Time	11.20	20.91	26.07	27.22	31.01	57
Avg Shots	2.65	6.11	7.70	7.84	9.64	13.65
Highest Level	3	7	8	8.82	10	19
Wins Rate	0.60	0.85	0.90	0.88	0.95	1
Max Score	196	429.20	512.50	536.40	630	1149
Total Time	221	418.20	521.50	544.60	624.20	1191

the adopted definition. The remainder 17 players are the ones that, although played 20 or more levels, did not answer the questionnaire. Probably, these are the ones that stopped playing when the questionnaire popped out and then came back to play again later. Nevertheless, most players were retained and played a considerable number of levels.

5.2 Dynamic Game Version Influences

The goal of this section is to present results that answer the remaining part of RQ2. Initially, the self-reported opinions of players from the experimental group are presented. Following, their in-game performance and behavior are shown.

A total of 265 players from experimental group answered the PX questionnaire. Similarly to the previous analysis, Table 6 demonstrates, for fun, *returnance* and curiosity as a whole, factors' statistics (average, median, and standard deviation), the Mu value used in the test to compare the answers' median, the test statistic (U), and the effect size. The table shows that all factors also had a large highly significant difference from the selected Mu, which represents the indifferent answer. Thus, we can conclude that this version had a positive influence on players as well, promoting fun, willingness to play the game again and curiosity.

Table 6 – Experimental group's self-reported experience.

PX Factor	Mean	Median	SD	Mu	U	Effect Size
Fun	4.430	5	0.868	3	27330.500*	0.832
Returnance	4.253	5	1.194	3	18109*	0.725
Curiosity	3.782	4	1.054	3	934000*	0.600

* $p < 0.01$

Table 7 presents how these players performed in this version until the point of answering the questionnaire. It demonstrates the summary (minimum, first quartile, mean, median, third quartile and maximum) of seven in-game metrics. Players from this version also differed in performance, having a substantially large interquartile (1st Q. - 3rd Q.) on most metrics. For instance, the wins rate ranged 50% in terms of minimum and

maximum value, while the average shot per level ranged from roughly 2 to 30, on average. Thus, the experimental group accomplished the experiment through distinct performances in terms of the analyzed in-game metrics.

Table 7 – Summary of experimental group's in-game performance.

Metric	Min.	1st Q.	Median	Mean	3rd Q.	Max
Avg Score	31.95	50.15	53.90	53.30	56.85	63.85
Avg Time	8.50	19.60	24.25	25.14	29.90	55.60
Avg Shots	2.10	5.25	7.25	7.79	9.37	30.10
Highest Level	3	5	7	7.54	9	20
Wins Rate	0.50	0.80	0.85	0.85	0.90	1
Maximum Score	183	338	442	465.50	558	1224
Total Time	170	392	481	501.20	598	1112

Concerning in-game behavior, an average of 59 levels (SD = 63) was played by the whole group (369 players). From this total, 78 did not answer the PX questionnaire and played less than 20 levels. Therefore, 79% of the ones who played this version can be considered retained by the testbed game. Following the same vein, the 26 that did not provide answers to the PX questionnaire, despite playing 20 levels or more, probably choose to not answer it and came back to play the game again later.

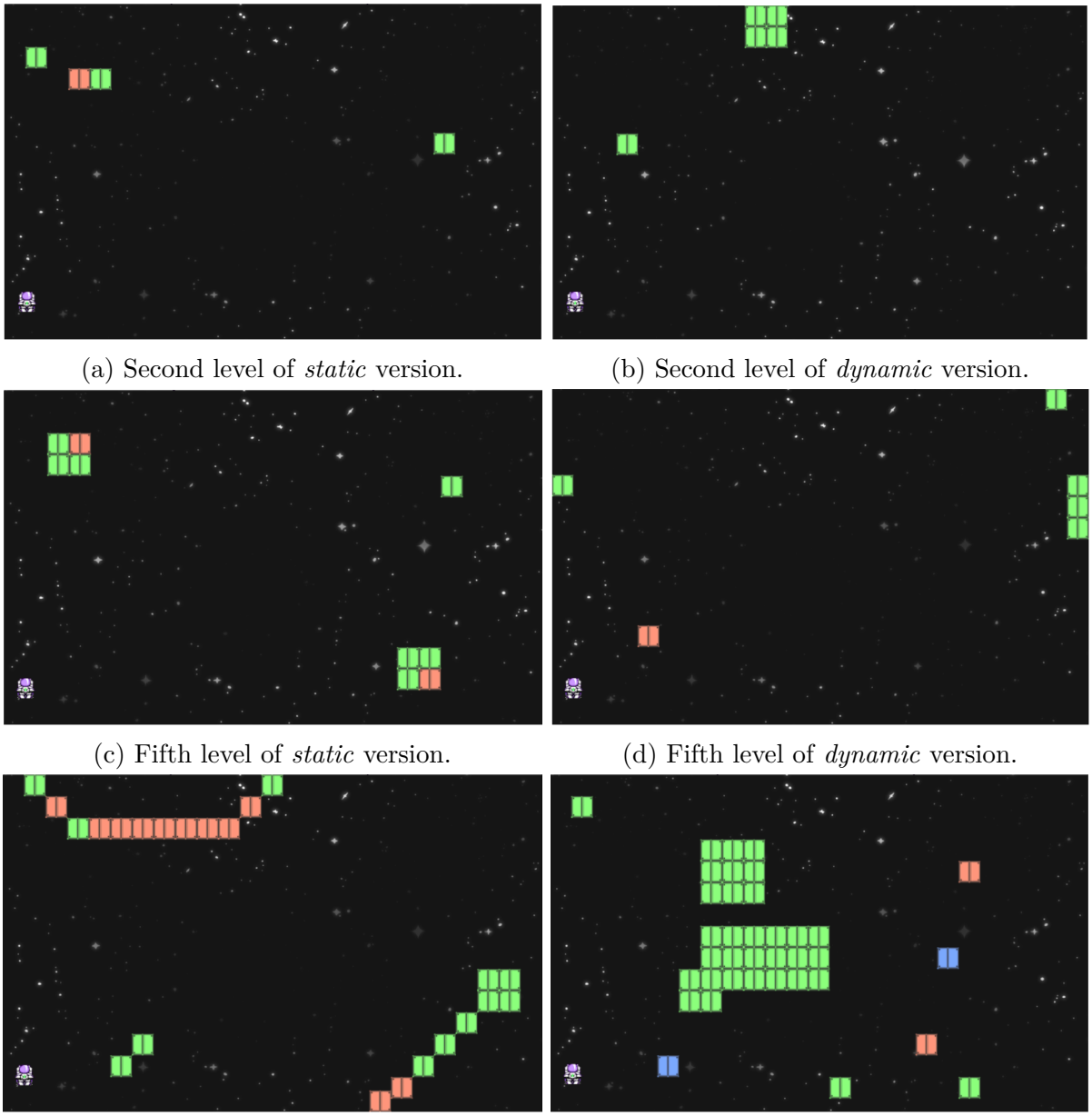
5.3 Dynamic Version vs Static Version

This section presents the results of the game's versions comparison. At first, a brief comparison in terms of level design is presented. Thereafter, groups self-reports, in-game behavior, and performance are compared. Finally, an analysis of demographic characteristics' impact on players opinions is conducted.

5.3.1 Levels Design

In Figure 4, three levels of each version are presented. On the left column, human-designed levels are demonstrated, whereas the right column displays the procedurally generated levels. We highlight that the math problem is not presented in these figures since they are created in the same fashion for both versions.

Samples of the second, fifth and tenth levels can be seen in Figure 4, which are shown by row. On the second and fifth levels, Figures 4a - 4d, there is no clear difference between versions, which might be due to the simplicity of the game. Few elements are present in each level, with boxes of different colors distributed along the level space. Differently, on the tenth level, Figures 4e and 4f, it is possible to notice their design differences. While the procedure of automatic level generation is based on placing single and rectangular blocks of boxes, the human-designer explored diverse placement strategies.



(a) Second level of *static* version.

(b) Second level of *dynamic* version.

(c) Fifth level of *static* version.

(d) Fifth level of *dynamic* version.

(e) Tenth level of *static* version.

(f) Tenth level of *dynamic* version.

Figure 4 – Randomly selected levels of both versions of *SpaceMath*.

As can be seen in Figure 4e, the human-authored design features stairs-like (bottom right) structures on this level. In contrast, the procedurally generated contains randomly placed boxes along with some quadratic blocks that end-up together, creating a larger rectangle with a small square block attached to it. In all, the key difference between versions is that, whilst the *dynamic* mainly depends on the PRNG to place elements in a way that form new structures, the *static* was originally designed with distinct ones by the human developer. In addition, the amount of elements is substantially different between versions, showcasing that there were variations within the game levels' design, despite *SpaceMath*'s simplicity.

5.3.2 Self-reports

Comparing players' self-reported experiences is this part's first step. It will demonstrate whether or not the way each group was influenced by its respective game version differed. Figure 5 illustrate their questionnaire responses with a boxplot of the nine PX factors: fun, *returnance* and the seven curiosity questions. As can be seen, for all factors, the distribution of responses from both groups was similar. In addition, Table 8 shows the average (SD) of each factor, along to results of the Kruskal-Wallis tests, demonstrating that the only significant difference was in C5 (I sought explanations for what I encountered in the game). Thereby, we can conclude that PCG promoted experiences as good as the human-generated content in eight out of nine factors according to players opinions.

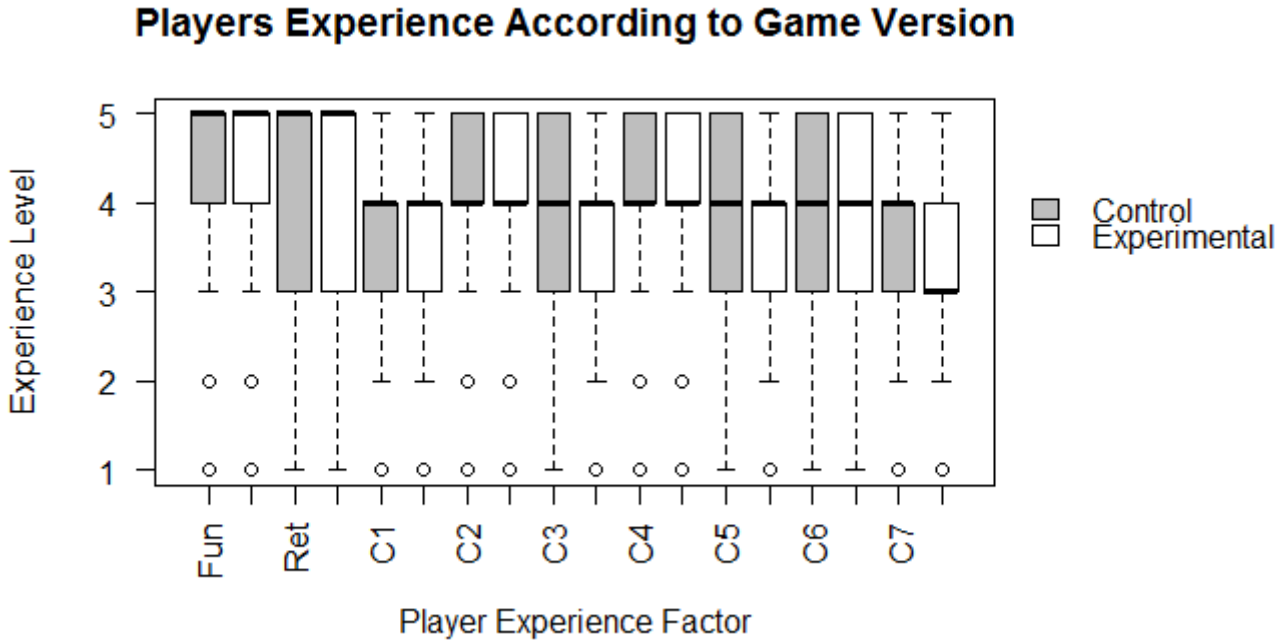


Figure 5 – Boxplot of PX for control and experimental group.

Then, we investigated C5 given that it was the only factor with a significant difference between groups. The goal was to visualize how players' answers differed. Figure 6 demonstrates the distribution of players' responses from both groups. It shows that players from the experimental group sought explanations for what they encountered less, in comparison to control's players. Thus, playing the *static* game version led players to look for explanations more than the *dynamic* one.

Furthermore, we also compared how players from both groups described their experiences. It was captured through 10 opposite attributes (see Chapter 4.1), where they could choose all the ones that they agreed with. Figure 7 shows their selections' distributions. It displays the five pairs in order (e.g. great - tiring, simple - boring), allowing us to see that positive attributes, mainly great and fun, were selected substantially more

Table 8 – Comparison of groups experience. Data represented as Mean (SD).

PXF	**Control**	**Experimental**	**KW-χ^2**
Fun	4.446 (0.778)	4.430 (0.868)	0.104
RET	4.397 (1.039)	4.253 (1.194)	1.373
C1	3.789 (0.982)	3.642 (1.043)	2.593
C2	4.227 (0.836)	4.155 (0.872)	1.028
C3	3.888 (0.897)	3.815 (0.917)	0.786
C4	4.161 (0.885)	4.042 (0.889)	3.166
C5	3.810 (1.041)	3.562 (1.123)	6.583*
C6	3.988 (0.979)	3.849 (1.077)	1.585
C7	3.450 (1.201)	3.411 (1.219)	0.161

* $p < 0.05$

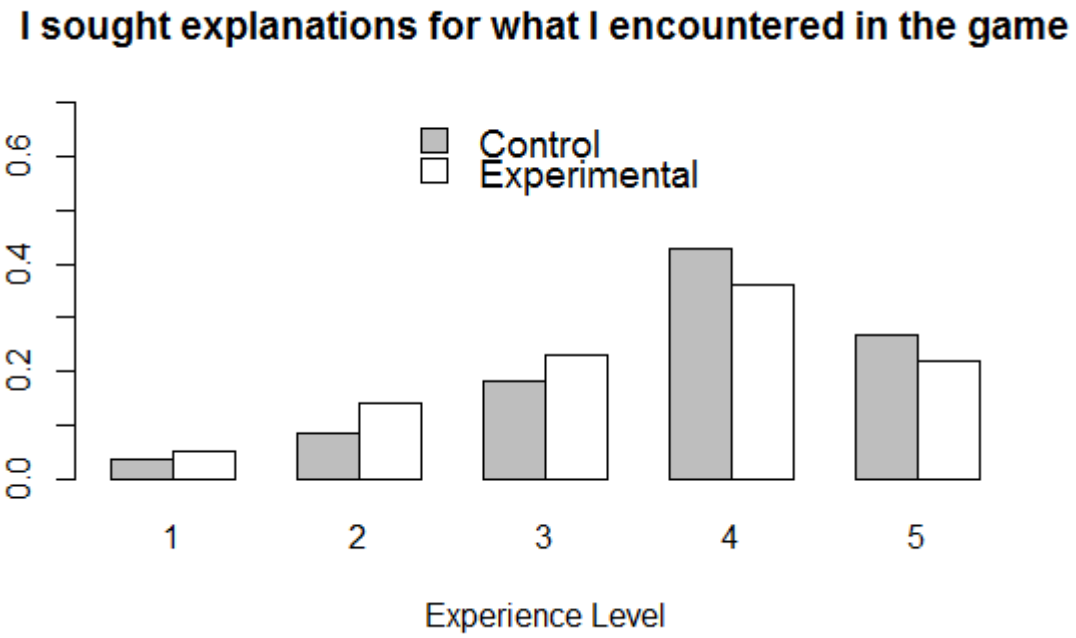


Figure 6 – Barplot of answers of the fourth curiosity question.

than the others for both versions. Also, it shows that opinions from both groups were similar, which was confirmed through χ^2 homogeneity test ($\chi^2 = 7.218$, df $= 9$, $p = 0.614$). Therefore, players described their experiences mostly using positive attributes, where the differences in the distribution of selected attributes were insignificant between groups.

5.3.3 In-Game Behavior

Game's versions were also compared on a different perspective, considering their in-game behavior. Players' retention, engagement, and performance, according to the notions adopted in this work, were analyzed to identify whether there exist some significant differences in playing one version or another.

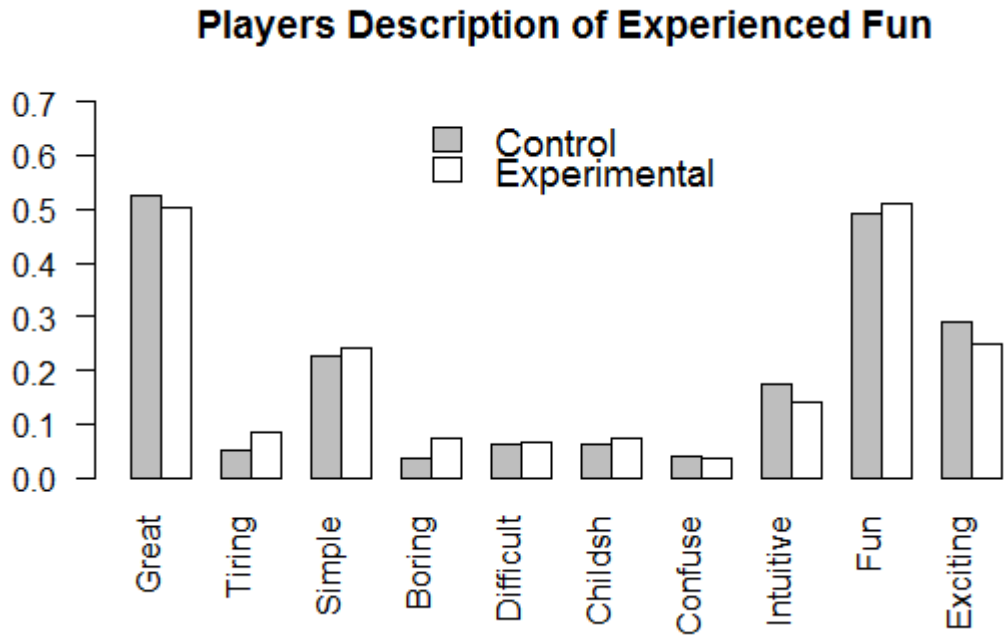


Figure 7 – Barplot of PX description.

Retention was evaluated considering players that played at least 20 levels and answered the questionnaire versus the ones that did not. From 355 players who interacted with the static version, 96 played less than 20 levels. Considering players from the dynamic version, 78 of the 369 did not play at least 20 levels as well. According to the χ^2 homogeneity test, this difference is insignificant ($\chi^2 = 1.877$, df = 1, p = 0.171). Thus, players who played the procedurally generated levels was as retained as the ones who played the human-designed levels.

Groups' engagement was assessed through the average of levels that each one played. This analysis was performed based on all players from both groups to investigate possible differences in their behavior in a general picture. Players from the control group played 52.72 levels on average (SD = 53.455), while players from experimental played an average of 58.73 levels (SD = 63.479). This difference was not significant according to the U test (U = 69254, p = 0.182). Hence, players from *dynamic* version was as engaged as players from the *static* one.

In addition, we also investigated the engagement of non-retained players only. Similarly, the differences between groups were insignificant (U = 3583.5, p = 0.628) according to the U test. For this situation, players from the static version played an average of 10.9 levels (SD = 5.214), whilst players from the dynamic played 10.51 on average (SD = 5.557). This finding corroborates with the indication that players from one version were engaged as players from the other, confirming it even for non-retained players.

Lastly, the performance that players yielded playing each version was compared.

Metrics related to their performance that were evaluated are: average score, highest level achieved, wins rate, the maximum score achieved and total time spent until reaching the questionnaire. Note that, differently from the other behavioral comparisons, only data from players who answered the questionnaire are used here. Also, these data are related to their gameplays before answering the questionnaire only. The goal is to investigate how players performed in each version since, after playing the twentieth level, they could end up playing a different game version.

The comparison of groups performance can be seen in Table 9. It shows the average (SD) for the five aforementioned metrics and the statistic results of the U test, denoting with an asterisk significant differences. It shows that players from the control group performed significantly better than players from experimental and took more time to reach the questionnaire. We claim that since answering the questionnaire is based on the amount of played levels, the small wins rate shows that players from experimental group had more lost and, consequently, spent less time playing game levels. Therefore, we can conclude that the *dynamic* version provided players with more challenger experiences than *static*.

Table 9 – Comparison of groups performance. Data represented as Mean (SD).

Metric	**Control**	**Experimental**	**U test**
Avg Score	54.741 (5.366)	53.304 (5.507)	37394.500*
Highest Level	8.822 (2.602)	7.540 (3.002)	41988*
Wins Rate	0.884 (0.064)	0.846 (0.08)	42162.500*
Maximum Score	536.430 (155.328)	465.453 (175.621)	41259.500*
Total Time	544.583 (175.822)	501.158 (151.263)	36587.500*

* $p < 0.05$

5.3.4 Demographics Influences

In terms of players self-reports, considering the whole sample of each group, there is only one factor that had a significant difference between game versions, C5. However, our sample is composed of a varied set of subjects with distinct characteristics. Thus, we compared control and experimental groups in terms of subsamples in order to identify if differences between them exist (e.g. control males vs experimental males).

Firstly, we compared self-reports according to different demographic classes, beginning with the boolean ones (i.e. genre, gamer and having internet). As can be seen in Table 10, no significant difference was found between male players from both groups, whereas there was a significant difference in a single factor, C5, for females. Similarly, gamer players had no difference in their experiences, while non-gamers significantly differed in C5 only, as demonstrated by Table 11. Unlike the previous demographics, subjects with internet access through a computer at home significantly differed whilst the ones without it had an insignificant difference in C5, which is shown in Table 12. Hence, we

have evidence that genre, considering itself a gamer or not, and whether a player has access to the internet at home through a computer influenced players seeking explanations for what they encountered in the game.

Table 10 – Comparison of groups experience according to each Genre. Data represented as Mean (SD).

	Males			Females		
PXF	**Control**	**Experimental**	**KW-χ^2**	**Control**	**Experimental**	**KW-χ^2**
Fun	4.405 (0.798)	4.384 (0.926)	0.091	4.532 (0.731)	4.500 (0.772)	0.002
RET	4.374 (1.055)	4.296 (1.172)	0.159	4.443 (1.01)	4.189 (1.228)	1.875
C1	3.810 (1.016)	3.616 (1.072)	2.692	3.747 (0.912)	3.679 (1)	0.200
C2	4.239 (0.815)	4.189 (0.894)	0.143	4.203 (0.883)	4.104 (0.839)	1.177
C3	3.951 (0.866)	3.843 (0.868)	1.272	3.759 (0.95)	3.774 (0.988)	0.023
C4	4.196 (0.881)	4.063 (0.939)	1.856	4.089 (0.894)	4.009 (0.811)	0.833
C5	3.810 (1.097)	3.654 (1.114)	1.945	3.810 (0.921)	3.425 (1.129)	5.350*
C6	4.080 (0.962)	3.843 (1.117)	3.197	3.797 (0.992)	3.858 (1.018)	0.328
C7	3.509 (1.239)	3.484 (1.195)	0.068	3.329 (1.118)	3.302 (1.251)	0.029

* $p < 0.05$

Table 11 – Comparison of groups experience according to being a gamer or not. Data represented as Mean (SD).

	Gamers			Non-gamers		
PXF	**Control**	**Experimental**	**KW-χ^2**	**Control**	**Experimental**	**KW-χ^2**
Fun	4.550 (0.743)	4.468 (0.935)	0.091	4.344 (0.801)	4.396 (0.804)	0.528
RET	4.533 (0.961)	4.254 (1.206)	3.573	4.262 (1.097)	4.252 (1.186)	0.032
C1	3.842 (1.061)	3.746 (1.095)	0.485	3.738 (0.898)	3.547 (0.987)	2.573
C2	4.200 (0.922)	4.167 (1.002)	0.008	4.254 (0.745)	4.144 (0.738)	1.903
C3	3.875 (0.975)	3.897 (0.937)	0.010	3.902 (0.817)	3.741 (0.896)	1.827
C4	4.258 (0.930)	4.079 (0.985)	2.785	4.066 (0.831)	4.007 (0.794)	0.641
C5	3.875 (1.042)	3.667 (1.193)	1.319	3.746 (1.041)	3.468 (1.052)	5.772*
C6	4.067 (0.932)	3.873 (1.207)	0.562	3.910 (1.020)	3.827 (0.947)	0.864
C7	3.633 (1.236)	3.595 (1.260)	0.043	3.270 (1.143)	3.245 (1.160)	0.099

* $p < 0.05$

Secondly, we investigated how the relationship from rank and numeric demographic variables differ between game versions, as can be seen in Table 13. It shows, for each group (G), Kendall's correlation coefficient from age, weekly playing hours, school type, likes math, and knows math to the nine self-reported PX factors. The farther the correlation is from 0 the more the dependent variable (PX) is affected by the independent variable (demographics). Thus, we can see that the experimental group (E) was less affected than the control (C) in terms of age. In contrast, the experimental group was more impacted by their affinity to math, while in terms of weekly playing time and school type the correlations were more similar and balanced between groups. Thereby, while the age of players from the *static* version impacted their experience more than the ones from *dynamic*

Table 12 – Comparison of groups experience according to having a computer with internet access at home or not. Data represented as Mean (SD).

	Has it			Has not		
PXF	**Control**	**Experimental**	**KW-χ^2**	**Control**	**Experimental**	**KW-χ^2**
Fun	4.447 (0.746)	4.430 (0.858)	0.161	4.444 (1.013)	4.435 (0.992)	0.013
RET	4.386 (1.039)	4.256 (1.184)	0.960	4.481 (1.051)	4.217 (1.313)	0.506
C1	3.791 (0.961)	3.628 (1.036)	2.763	3.778 (1.155)	3.783 (1.126)	0.001
C2	4.200 (0.833)	4.161 (0.861)	0.224	4.444 (0.847)	4.087 (0.996)	2.38
C3	3.902 (0.873)	3.831 (0.893)	0.749	3.778 (1.086)	3.652 (1.152)	0.093
C4	4.177 (0.846)	4.054 (0.879)	2.593	4.037 (1.16)	3.913 (0.996)	0.703
C5	3.819 (1.023)	3.599 (1.108)	4.621*	3.741 (1.196)	3.174 (1.230)	2.976
C6	3.991 (0.981)	3.855 (1.054)	1.587	3.963 (0.980)	3.783 (1.313)	0.035
C7	3.447 (1.202)	3.430 (1.211)	0.038	3.481 (1.221)	3.217 (1.313)	0.562

* $p < 0.05$

which, on the other hand, had experiences more related to their affinity to math, the other demographic factors were nearly balanced between versions.

Table 13 – Correlation from demographics to PX for each group.

G	**Fun**	**RET**	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**
					Age				
C	-0.248*	-0.262*	-0.295*	-0.205*	-0.235*	-0.250*	-0.114*	-0.196*	-0.267*
E	-0.158*	-0.197*	-0.227*	-0.163*	-0.234*	-0.151*	-0.074	-0.175*	-0.186*
					Weekly Playing Hours				
C	0.045	-0.062	-0.069	0.002	-0.037	0.045	-0.001	-0.032	0.017
E	-0.056	-0.061	-0.046	-0.007	-0.014	-0.064	-0.024	-0.088	0.033
					School Type				
C	0.006	0.017	-0.001	0.007	0.092	0.089	0.025	0.023	0.079
E	-0.021	-0.065	-0.015	-0.032	0.053	-0.010	-0.084	0.035	0.018
					Likes Math				
C	0.105	0.122*	0.203*	0.125*	0.071	0.183*	0.141*	0.106*	0.205*
E	0.248*	0.230*	0.199*	0.215*	0.168*	0.225*	0.181*	0.160*	0.257*
					Knows Math				
C	0.092	0.040	0.089	-0.001	-0.026	0.040	-0.010	0.004	0.099
E	0.160*	0.097	0.048	0.180*	0.125*	0.143*	0.079	0.137*	0.114*

* $p < 0.05$

Furthermore, how subjects' characteristics influenced their performance was analyzed as well. In contrast to players' opinions, there were significant differences in all performance metrics. Then, in these further analyzes, we sought for the cases wherein the difference was insignificant. In terms of average score, both females and non-gamers did not significantly differ, while males and gamers did not differ on the total time, as can be seen in Tables 14 and 15. On the remaining metrics, there were significant differences in these four subsamples of both groups. In addition, as can be viewed in Table 16, for players that do not have internet access through a computer at home, there was no

significant difference in all metrics. On the other hand, the table displays that, for the remaining players, all differences were significant.

Table 14 – Comparison of groups performance according to genre. Data represented as Mean (SD).

	Males			Females		
Metric	**Control**	**Experimental**	**U test**	**Control**	**Experimental**	**U test**
Avg Score	56.058 (4.771)	54.775 (5.363)	14976.500*	52.022 (5.530)	51.099 (4.978)	4776.500
H. Level	9.135 (2.723)	8.201 (3.266)	16044*	8.177 (2.212)	6.547 (2.226)	5925*
Wins Rate	0.892 (0.061)	0.862 (0.078)	16336*	0.869 (0.067)	0.822 (0.077)	5762.500*
Max Score	564.896 (161.379)	509.491 (190.011)	15953.500*	477.696 (123.564)	399.396 (126.007)	5702*
Total Time	497.865 (149.450)	472.799 (145.441)	14346	640.975 (187.495)	543.698 (150.509)	5378.500*

* $p < 0.05$

Table 15 – Comparison of groups performance according to being a gamer or not. Data represented as Mean (SD).

	Gamers			Non-gamers		
Metric	**Control**	**Experimental**	**U test**	**Control**	**Experimental**	**U test**
Avg Score	55.598 (5.225)	53.707 (5.572)	9151.500*	53.897 (5.390)	52.939 (5.441)	9507
H. Level	9.092 (2.819)	7.690 (3.194)	9800.500*	8.557 (2.350)	7.403 (2.823)	11199*
Wins Rate	0.888 (0.063)	0.846 (0.084)	9984.500*	0.880 (0.065)	0.846 (0.077)	11063.500*
Max Score	560.083 (168.254)	476.484 (184.125)	9746*	513.164 (138.246)	455.453 (167.582)	10868*
Total Time	515.275 (161.619)	479.563 (145.178)	8597	573.41 (184.912)	520.734 (154.494)	9811*

* $p < 0.05$

Table 16 – Comparison of groups performance according to having a computer with internet access at home or not. Data represented as Mean (SD).

	Has it			Has not		
Metric	**Control**	**Experimental**	**U test**	**Control**	**Experimental**	**U test**
Avg Score	55.127 (5.123)	53.588 (5.376)	30747*	51.666 (6.313)	50.319 (6.086)	356.500
H. Level	9.033 (2.611)	7.541 (2.887)	34978*	7.148 (1.834)	7.522 (4.111)	346.500
Wins Rate	0.891 (0.058)	0.847 (0.079)	35377.500*	0.831 (0.082)	0.830 (0.090)	316
Max Score	548.921 (155.276)	466.847 (169.010)	34434.500*	436.963 (116.709)	450.783 (238.663)	351.500
Total Time	542.814 (177.052)	492.227 (149.201)	30409.500*	558.667 (168.223)	595.130 (143.518)	250.500

* $p < 0.05$

Moreover, the relationship between rankable demographic characteristics to performance is shown in Table 17. It demonstrates Kendall's correlation coefficient from age and weekly playing hours to average score, highest levels, wins rate, maximum score and total time for each group (G). Other demographics were not reported because the correlations were insignificant to all performance metrics. The table shows that the performance

of both experimental (E) and control (C) groups were affected by these two demographics. However, there is not a group that is clearly more affected by one characteristic than the other. What can be seen is that there is a balance, with one group being more impacted in some variables while the other in more impacted by the remaining. Hence, the demographics influence on players performance were more balanced between groups than dominant in one or another.

Table 17 – Correlation from demographics to players performance for each group.

G	Avg Score	H. Level	Wins Rate	Max. Score	Total Time
			Age		
C	0.404*	0.217*	0.191*	0.291*	-0.359*
E	0.388*	0.297*	0.323*	0.331*	-0.214*
			Weekly Playing Hours		
C	0.191*	0.121*	0.113*	0.158*	-0.159*
E	0.217*	0.153*	0.214*	0.165*	-0.074

* $p < 0.05$

In sum, performing further investigations we were able to identify evidence suggesting from where the differences between groups in their self-reports (C5) and in-game behavior (performance) came. For the former, the significant difference in C5 is existent for females, gamers, and players who have internet access through a computer at home. In contrast, performance was mostly dissimilar, wherein its insignificant differences originated mainly by not having a computer with internet access at home.

In addition, we found that players from the experimental group had their experiences less dependent from their age, whereas they were more related to players affinity with math than the other group. On the other hand, the impact of weekly playing hours and school type on PX were nearly balanced. Similarly, most demographic attributes were insignificantly correlated to players performance, whereas the ones that were had no clear difference in their impacts on each group.

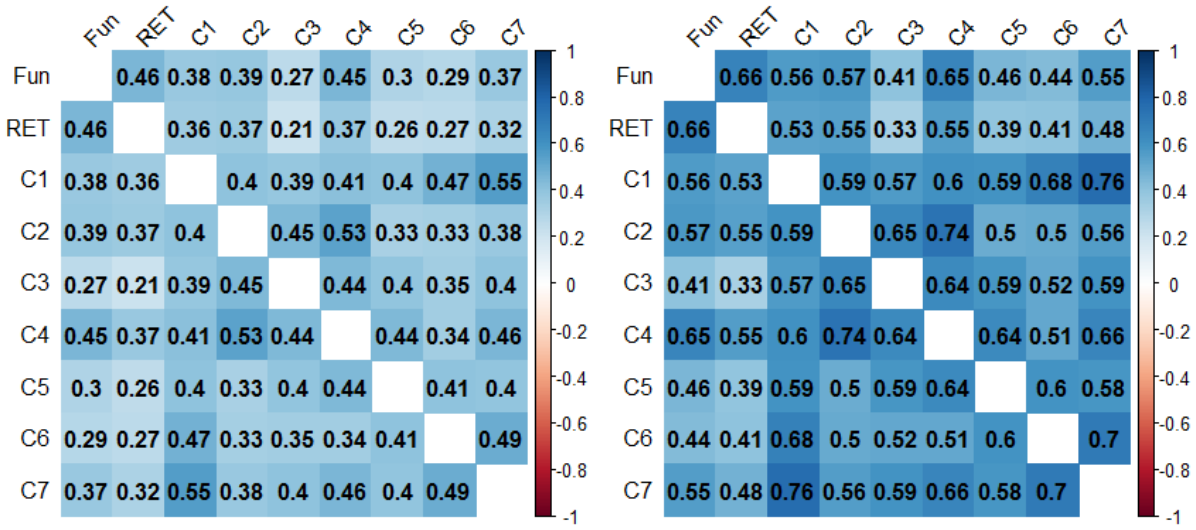
5.4 Correlation Between Player Experience Factors

Answering RQ5 is the focus of this section. At first, we investigated correlations to the curiosity as a whole, considering the seven question's average for each player. Then, we evaluated it to each curiosity statement separately.

In terms of interaction from fun to curiosity, a moderate to strong correlation was found, which is highly significant ($\tau = 0.431$, $p < 0.001$, $r = 0.626$). Similarly, considering *returnance*, a moderate to strong correlation that is also highly significant was found ($\tau = 0.373$, $p < 0.001$, $r = 0.553$). These positive correlations indicate that as players have more fun and intent to return more, they are more curious as well. Thus, we can conclude

that players' experienced fun and willingness to play the game again are highly correlated to their curiosity, having moderate to strong impacts on it.

In order to identify how each curiosity factor is correlated with the others, Figure 8 displays a correlation matrix. The order of curiosity questions is the same as the one in Section 4.1. As can be seen in the figure, all correlations are positive, indicating that as one of them increases the others tend to increase as well, and highly positive ($p < 0.001$) according to Kendall's test. Also, the degree of interaction between these factors ranges from moderate (0.33) to strong (0.76) according to τ coefficient converted to the r coefficient.



(a) Kendall's coefficient.

(b) Kendall's coefficient converted to Pearson's.

Figure 8 – Degree of correlation between PX factors.

These findings demonstrate that players who report better experiences (i.e. fun and *returnance*) are more likely to say that they were curious about the game itself and its educational content. Hence, for instance, improving their fun is expected to have a large impact on their curiosity as a whole, or a moderate to large influence on their motivation to learn more about math from the game (C1). Moreover, the findings also show that all curiosity statements are significantly correlated, ranging between small and strong interactions.

5.5 Characteristics Related to Players Experience

This section answers RQ4. Firstly, we investigated if each demographic characteristic is related to each PX. Then, this procedure was replicated in order to investigate relationships between players' performance and their reported experience.

The degree of correlation - τ (r) - from six demographic attributes to the three PX factors are demonstrated in Table 18, wherein significant relationships are denoted by an asterisk. As can be seen, age was the most correlated characteristic to the three factors, with a moderate interaction, whereas these interrelationships are small for the remaining characteristics, with exception of school type to the three, knows math to *returnance*, and likes math to curiosity. This provides significant evidence that most ordinal and continuous demographic characteristics have small to moderate correlations to PX, but not about dichotomous.

Table 18 – Correlation degree from demographics to PX factors.

Attribute	**Fun**	***Returnance***	**Curiosity**
Age	-0.199 (-0.307)*	-0.226 (-0.347)*	-0.226 (-0.347)*
School Stage	-0.157 (-0.244)*	-0.123 (-0.192)*	-0.186 (-0.288)*
Weekly Playing Hours	-0.010 (-0.015)	-0.062 (-0.098)	-0.018 (-0.029)
School Type	-0.006 (-0.010)	-0.028 (-0.044)	0.021 (0.033)
Likes Math	0.181 (0.280)*	0.173 (0.269)*	0.215 (0.331)*
Knows Math	0.131 (0.204)*	0.069 (0.108)	0.095 (0.148)*

*p < 0.05

The remaining demographic attributes are dichotomous, thereby we investigated whether they are associated with PX through the Chi-Square independence test. The analyzed hypothesis is that there exists an association between the experience of players of each class from these characteristics. Tables 19, 20 and 21 present these analyses results for fun, *returnance* and curiosity, respectively. In each table, the dichotomous attributes are genre, having a computer with internet access at home and considering itself a gamer. Answers are abbreviated as follows: completely disagree (CD), disagree (D), indifferent (I), agree (A), and completely agree (CA). Lastly, the test statistic is shown (χ^2), indicating significance through an asterisk. Note that Table 21 has seven times more answers because it considers the seven curiosity questions.

Table 19 – Independence test of Fun and dichotomous demographics.

Att	**CD**	**D**	**I**	**A**	**CA**	χ^2
Male						
No	1 (0.005)	2 (0.011)	17 (0.092)	46 (0.249)	119 (0.643)	3.181
Yes	6 (0.019)	4 (0.012)	33 (0.102)	93 (0.289)	186 (0.578)	
Computer with internet access at home						
No	2 (0.04)	1 (0.020)	3 (0.060)	11 (0.220)	33 (0.660)	4.816
Yes	5 (0.011)	5 (0.011)	47 (0.103)	128 (0.280)	272 (0.595)	
Gamer						
No	1 (0.004)	4 (0.015)	35 (0.134)	78 (0.299)	143 (0.548)	15.070*
Yes	6 (0.024)	2 (0.008)	15 (0.061)	61 (0.248)	162 (0.659)	

* p < 0.05

Table 20 – Independence test of *Returnance* and dichotomous demographics.

Att	**No**	**Maybe**	**Yes**	χ^2
		Male		
No	9 (0.049)	47 (0.254)	129 (0.697)	0.137
Yes	14 (0.043)	79 (0.245)	229 (0.711)	
Computer with internet access at home				
No	3 (0.060)	10 (0.200)	37 (0.740)	0.878
Yes	20 (0.044)	116 (0.254)	321 (0.702)	
		Gamer		
No	12 (0.046)	73 (0.280)	176 (0.674)	2.877
Yes	11 (0.045)	53 (0.215)	182 (0.740)	

* $p < 0.05$

Table 21 – Independence test of Curiosity and dichotomous demographics.

Att	**CD**	**D**	**I**	**A**	**CA**	χ^2
			Male			
No	42 (0.032)	116 (0.090)	257 (0.198)	561 (0.433)	319 (0.246)	22.063*
Yes	81 (0.036)	150 (0.067)	435 (0.193)	884 (0.392)	704 (0.312)	
Computer with internet access at home						
No	24 (0.069)	24 (0.069)	63 (0.180)	132 (0.377)	107 (0.306)	14.814*
Yes	99 (0.031)	242 (0.076)	629 (0.197)	1313 (0.410)	916 (0.286)	
			Gamer			
No	49 (0.027)	147 (0.080)	408 (0.223)	791 (0.433)	432 (0.236)	64.900*
Yes	74 (0.043)	119 (0.069)	284 (0.165)	654 (0.380)	591 (0.343)	

* $p < 0.05$

As can be seen in Table 19, considering itself a gamer was the only characteristic associated to players fun. Differently, their *returnance* was not associated to none of the dichotomous demographic attributes, as shown in Table 20. On the other hand, players' curiosity had an association to genre, being a gamer and having a computer with internet access at home, which is demonstrated in Table 21. These findings show that, while curiosity is associated with the three dichotomous attributes, *returnance* is independent of them, whilst fun depends on being a gamer only.

The correlation from seven in-game metrics to the three PX factors are shown in Table 22 in the form of τ (r), wherein significance is denoted by an asterisk. As can be seen in the table, all performance metrics had small but significant correlations to the PX factors, with exception of average shots to curiosity. Time-related metrics (average time and total time) were the only characteristics that had positive relationships. This suggests that spending more time to reach the questionnaire, which indicates that the gameplay was more challenging, was perceived as better experiences. This corroborates with the negative interaction from the average score to PX, for instance, which implies that smaller average scores were perceived as better experiences as well. Thus, there

exists a small interrelationship between players' performance and opinions regard their experience.

Table 22 – Correlation degree from performance to PX factors.

Metric	Fun	*Returnance*	Curiosity
Average Score	-0.144 (-0.225)*	-0.164 (-0.255)*	-0.175 (-0.271)*
Average Time	0.090 (0.141)*	0.099 (0.154)*	0.098 (0.154)*
Average Shots	-0.071 (-0.111)*	-0.082 (-0.128)*	-0.060 (-0.094)
Highest Level	-0.085 (-0.133)*	-0.129 (-0.201)*	-0.073 (-0.114)*
Wins Rate	-0.089 (-0.140)*	-0.126 (-0.197)*	-0.121 (-0.189)*
Maximum Score	-0.112 (-0.176)*	-0.139 (-0.217)*	-0.102 (-0.160)*
Total Time	0.094 (0.147)*	0.106 (0.165)*	0.105 (0.164)*

*$p < 0.05$

5.6 Summary of Main Findings

RQ1: How does the *static*/*dynamic* game version influence players opinions and performance?

Our findings showed that, concerning their self-reports, both game versions provided positive experiences, made players curious and were mostly described with positive attributes. Also, the experiments have shown that over 50 levels were played on average and that over 70% of them were retained. In terms of their performance, both groups had a high rate of wins, as it is expected by the game's mechanics, with a slightly better performance to the control group. Therefore, we conclude that the two versions influenced players positively, retaining and making most of them curious and engaged to play several levels with good performances.

RQ2: Does the *static* and *dynamic* game version differ in terms of how they influence players?

The difference between the experiences from each version was insignificant in all self-reported factors but one, the *I sought explanations for what I encountered in the game* (C5) statement. Further investigating this question, we found that comparing the answers of each version, this difference was significant only for: female; gamers; and the ones with internet access through a computer at home. Also, the influence of age was strongest on the ones who played the *static* version, whilst the remainder relied more on their affinity with math.

Moreover, considering in-game data, there was a significant difference in their performance, whereas their behavior's (retainment and engagement) differences were insignificant. Groups performance were mostly significantly different, wherein the most impacting factor was having internet access at home from a computer. Thus, game versions

differed in only one of the nine self-reported factors assessed and in players' performance, wherein demographic attributes had shown insights from where these differences emerged.

RQ3: Are players experienced fun and willingness to play the game again interrelated to their curiosity?

The experiments provided evidence that considering both groups, players' fun and *returnance* have a strong positive correlation to their curiosity as a whole, which is highly significant. Further analyzing how these factors are correlated to each curiosity question, we found that they are significantly correlated to them all, with a degree of positive interrelation that ranges from moderate to strong. In addition, we had evidence that fun and *returnance* also have a strong positive and highly significant correlation. Hence, players' self-reports of their experience, such as fun and willingness to play the game again, are interrelated to their curiosity as a whole and to its factors separately, with a degree that ranges from moderate to strong.

RQ4: Are players demographics/in-game performance related to their experience?

Our experiments demonstrated that ordinal and continuous demographic attributes (e.g. school stage and age) have small to moderate negative significant correlations to PX. In contrast, players' affinity with math has a small positive correlation to the three PX factors, which is significant as well. On the other hand, whilst curiosity is associated with genre, being a gamer and having internet access through a computer at home, *returnance* is not, while fun depends of being a gamer only.

Furthermore, players' performance metrics are also significantly correlated to their experience. The only exception is the average of shots per level not having a significant correlation to curiosity. Nevertheless, all relationships have a small negative degree. Thereby, most demographic attributes and in-game metrics are whether correlated or associated with PX, even though these interactions are mostly small.

6 DISCUSSION

Our testbed game, *SpaceMath*, demonstrated to be a reliable tool to induce players to practice the four basic operations of math. They reported positive outcomes in all questionnaire statements, besides describing their experiences mostly as positive. Thus, practitioners on the use of this kind of tool, such as teachers, can take advantage of its qualities on activities with students. Also, since it approaches a subject which is fundamental for almost all other fields of math, it can be used by other audiences as well, such as teenagers, younger and others. Its capacity to always providing new contents (i.e. levels and math problems) is its main feature, which collaborates to prevent players from having to face repeated experiences and playing contents already seen. Hence, *SpaceMath* is a valuable tool which encourages its players to practice math in a playful fashion, promotes real-time generated content and leads to positive experiences, thus contributing to math education.

Additionally, although most of *SpaceMath*'s levels are algorithmically generated, it roughly influenced PX. In comparison to human-designed levels, the generic level generation was able to promote equivalent experience levels in all but one factor. Through a deeper analysis, seeking to identify possible reasons influencing this single significant difference, we found that it vanished on some subsamples. These subsamples were divided according to players' demographics, showing that considering only males, gamers or the ones without internet access through a computer at home, groups' reports about their experiences were practically the same. This finding demonstrates that players with different backgrounds are influenced in different ways by PCG. While some perceive their experience as good as if they were playing human-designed levels, others face a small difference in terms of *seeking explanations for what they encountered in the game* (C5). Therefore, our experiment has shown that PCG provided experiences equivalent to the human-authored levels in nearly all self-reported factors and insights about which characteristics play a role in it.

However, there are some threatens that emerge from our experiment, such as human bias and the game design. Even though human-authored content can represent a bias, mainly if poorly designed [25], we argue that the high self-reported experiences remedy this threat. As shown, most players reported positive experiences and claimed intent to play *SpaceMath* again, besides getting curious about it. Yet, if the game's feedback were aligned wherein the best possible score was 100 rather than 74 or if it provided another scale such as stars, players could understand their performance more easily, be more motivated by the game, and, possibly, improve their performance and experience. Nevertheless, our findings show that developers may take advantage of PCG benefits in their games

without jeopardizing the game's outcomes. However, the performance of players from the *static* version was better than the ones from the *dynamic*. Although this metric's link to the self-reported experiences was small, control is a desirable property to PCG algorithms [50] and could mitigate performances differences.

Despite assessing the algorithm's controllability is not the focus of this work, designing the level generation to produce easier (or harder) levels is possible in spite of the simplicity of the method applied. The main algorithm's parameter is based on the player's winning streak, which is expected to control levels' difficulty. Figure 9 displays the relationship of players' winning streak to the average score in each streak. As can be seen, there is a strong negative correlation ($r = 0.901$, $p < 0.001$), supporting the assumption that increasing this parameter will decrease players' performance, thus increasing levels' difficulty. Hence, it is possible to control the game progression difficulty through the adjustment of this parameter in order to provide a challenging level comparable to the *static* version.

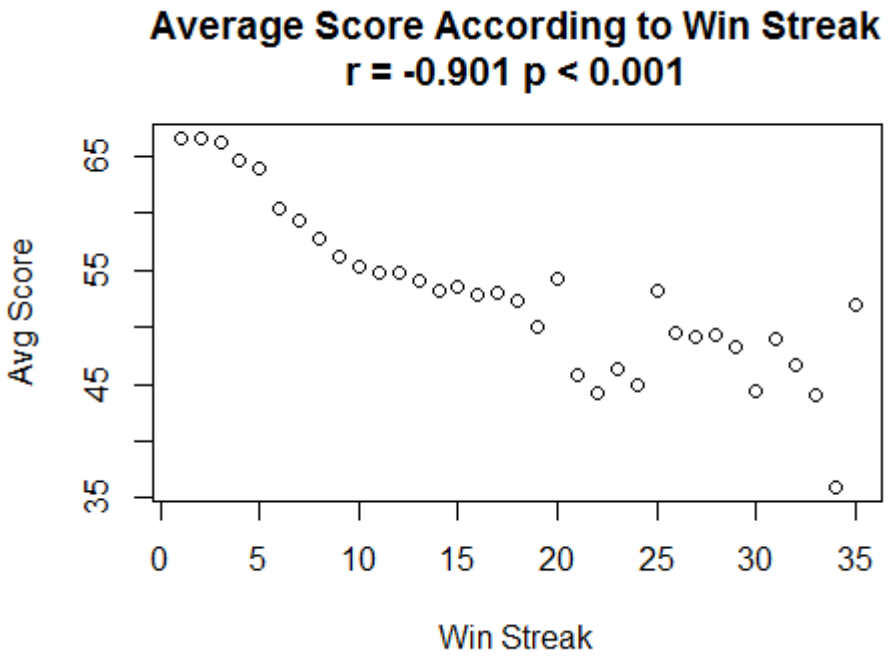


Figure 9 – Scatterer plot of win streak's impact on average score.

Nonetheless, although the majority of players claimed they were willing to play *SpaceMath* again, most of them still did not play it again at the time of writing this document. Considering all in-game data, differently from the experiments previously presented that used data until the point of answering the questionnaire only, we designed Figure 10. It shows the number of players (Y-axis) that played each amount of sessions (X-axis), wherein starting to play the game after a pause of at least one hour is considered a new session. As can be seen on the figure, over 85% (619 subjects) of our sample played

a single session. Thereby, despite indicating willingness to play *SpaceMath* again, most players did not yet, at the time of performing the analysis of this document.

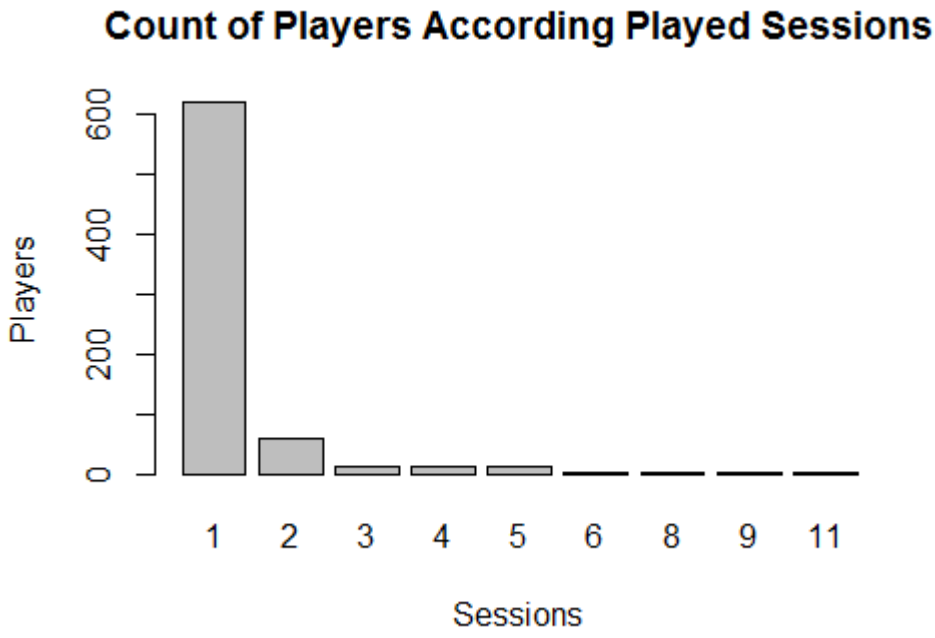


Figure 10 – Barplot of the amount of played sessions.

Crossing game data with players' opinions showed an inconsistency in terms of their *returnance*, which might emerge due to some reasons that are following discussed. Firstly, it might be the case that players were really willing to play the game again but then changed their mind or did not have the opportunity yet. For instance, they would play it again in a school activity, when there are limited options, but would not play it at home, on the other way. Another fact might be players being aware that they were participating in a research, which could bias them to report a positive (yes) answer even though they were not willing at all. Lastly, there is the fact that claiming willingness to play it again do not, necessarily, implies that they actually will do it.

In addition, we highlight that comparing sessions between game versions was not performed since considering all in-game data, players could have changed from *static* to *dynamic* due to the limited number of human-designed levels. Thus, it would be an unfair comparison. Also, while it is possible to *cross-validate returnance* with game data through how many of them actually did it, for other factors such as fun and curiosity this is not possible. Hence, most players reported that would play the game again but did not play a second session. Whilst it demonstrates an inconsistency between their intents and their behavior, it does not provide enough concerns to conclude that their self-reports were unreliable.

In sum, our analyses' results demonstrate that designers could adapt the gene-

rator's parameters to achieve the desired difficulty progression in similar applications. Hence, we contribute by showing that it is possible to use PCG, improving the game's development process, and still promote experiences nearly as good as the ones of human-authored contents, besides controlling content's difficulty. Also, we introduce the use of curiosity as an evaluation metric, which has not been used on previous researches of this kind [22, 43, 44]. In addition, we believe that these results can generalize to similar games, considering that PCG, for instance, increases their replay value [22], besides constantly providing new contents as well as the benefits shown by our experiments. Despite that, there is still the need for performing similar researches in order to validate this belief.

Furthermore, we found that players curiosity is highly correlated to their self-reported fun and willingness to play the game again. Mainly, researchers are concerned with these latter factors, while the former is especially valuable to serious games. The strong positive relationship between these factors allows designers to seek for improving PX in the *traditional* way and, thus increasing players' curiosity regarding the game itself and its serious content, based on our findings. Thereby, we also contribute by providing empirical evidence in terms of the assumption that improving players' fun and *returnance* has a strong impact on their curiosity.

Moreover, our analyses also demonstrated that players' demographics do influence their in-game experiences. Investigating how players in-game behavior/performance influence their experiences is common, while their demographic attributes are scantly used in these procedures. Our results presented evidence that characteristics such as age, school stage and considering itself a gamer have small to moderate relationships to their self-reports, whilst in-game metrics are also correlated to these factors. Therefore, it is valuable for player modeling researchers to also explore features based on this kind of data when creating their models, as they are related to PX. Thus, providing evidence that demographics are relevant to PX, besides in-game data, is another contribution of this study.

6.1 Literature Comparison

The evaluation of our research mainly focused on identifying the impact of PCG on PX. Thus, we expand the literature based on studies reviewed in Section 2.6, which have their key characteristics summarized in Table 23. As can be seen on the table, two works investigated level generation, besides ours, while the other created level's reefs.

Korn et al.[22] used a sample of 41 adult players which played both game versions and responded to a questionnaire mostly focused on reefs aesthetics. The others evaluated their approaches through a two-samples method, wherein each sample played a single game version. Connor, Greig e Kruse[43] assessed subjects regards their immersions and, similar

Table 23 – Comparison of researches on PCG's impact.

Ref	Year	Content	Age	N	Behavior	Opinion	Design
[22]	2017	Reef	18-≈45	41	-	aesthetic, preference	one-sample
[43]	2017	Level	18-35	20	-	immersion	two-samples
[44]	2015	Level	-	2377	engagement	-	two-samples
Ours	2018	Level	7-72	506	engagement, retention	experience, curiosity	two-samples

to Korn et al.[22], performed it based on adult players' opinions. In contrast, Butler et al.[44] focused on players' engagement, which was captured by their in-game behavior (i.e. time and amount of levels played).

Also, Butler et al.[44] was the only research where demographic data from players were not captured. Their game was hosted online, in a second party environment where players did not have to register to play. Thus, they did not have access to this type of data in their analysis. In spite of that, their approach featured a substantially higher number of participants (2377) in comparison to the others (20 and 41). Similarly, our testbed is hosted online, however, registering is a requirement to play it. Thereby, we had access to players demographics.

Moreover, we captured players' opinions as well, allowing us to analyze it along with their in-game behavior, differing from the three works. In addition, besides investigating players' entertainment factors (e.g. fun and *returnance*), we also examined players' curiosity, which is valuable to our serious application and had not been performed in this context yet. Our application itself is another differential, the usage of a DMG focused on arithmetic operations, since the only paper that used a game with serious purposes [44] was focused on fractions and did not evaluate any factor related to the serious subject.

In terms of sample size, we had substantially more data than two researches [22, 43], but substantially less than the third [44]. Nevertheless, we argue that this size should be enough to provide reliable findings and mitigate small sample biases. Despite that, our sample's characteristics are contrasting to others. While literature works are only formed by young and adults, when this type of info is available, ours is composed of children as well. Hence, we are able to investigate PCG impacts on a more varied sample and identify how it differs in e.g. children, teenagers, young and adults.

Lastly, the findings of our research are in line with the literature. Despite that, Korn et al.[22] differs from all similar researches, their subject of matter was game reefs rather than game levels. Hence, we argue that investigating a different type of game content is, probably, the rationale behind they finding results contrary to other studies. Differently, in comparison to works focused on the impacts of procedurally generated levels [43, 44], our work delivers similar results. As well as Connor, Greig e Kruse[43],

we found that playing on the *dynamic* game version provided nearly equivalent levels of experience. Similarly, our results also demonstrated that players from both game versions played insignificantly different numbers of levels, like the results of Butler et al.[44].

6.2 Limitations and Future Directions

There is the critique regard using a human in the content generation process because they might insert a bias on the results [25]. For instance, if the developed levels are of poor quality, this could favor the procedurally generated content and, thus jeopardize the findings. One way to remedy this problem could be to use more than one human-designed version. Thereby, it would be possible to identify whether some of them are better than others and, then use it as the baseline comparison. Another alternative is to also use a completely random generator, which would enable the comparison of whether both human-authored and the constructive approach to level generation excels the random content [44]. In this way, human biases would be mitigated, improving the reliability of the study's findings.

In this work, levels are generated through a simple and straightforward technique, the *constructive* method. It has been shown that more complex approaches, such as search-based ones, are preferred by players in comparison to both constructive [65] and randomly created content [84]. Therefore, performing similar researches where human-created content are compared to those techniques is expected to show that PCG can overcome the human-authored content. However, these techniques require the development of a fitness function, which is the most complex task of using them [35]. Thus, successfully employing it will mainly depend on designing a fitness function that provides players with game levels that are accordingly to their expectations. Moreover, there is a trade-off in terms of speed, whereas these are optimization algorithms that tend to be more costly than the constructive ones. Hence, employing search-based PCG has the potential to improve PX, despite it is harder to develop and computationally more costly.

Based on the aforementioned, the community would benefit from a public testbed, where other researchers could implement their PCG solutions and test them in the same way. Consequently, it would be more clear how different algorithms impact players since they could be tested using the same game and baseline (human-designed content), besides the same methodology. In addition, although there has been some research concerning what is the influences of PCG on PX, this specific field is yet emerging and demand more research [22, 43]. Thus, it is necessary to perform similar researches using different genres of games (e.g. platforms), samples (e.g. elderly) and goals (e.g. learning gains). Considering this context, the development of a public testbed for PCG algorithms would aid on the development of clearer and more reliable concerns regarding PCG's influences on players.

Additionally, in spite of the small to moderate correlations from in-game performance and players' demographic to their experiences, they still provide insights regarding how these characteristics affect PX. However, there is no guarantee that players provided their demographic attributes (e.g. age or likes math) truly, even for data captured through supervised applications. In spite of that, capturing larger amounts of data is expected to mitigate these noises, which is a common problem in terms of game data [31, 85]. Nevertheless, a player model could be developed based on these characteristics along to in-game data, in order to model their curiosity, for instance. This data-driven approach could also aid on remedying the dataset's noise through machine learning techniques, improving the model's performance. Then, it could be used as the fitness function of a search-based approach, driving the level generation towards improving PX, implementing the experience-driven PCG perspective [40]. Thereby, this adaptive game version could drive players curiosity, improving game's benefits to players from the educational perspective even more, while providing more tailored experiences.

Lastly, there is the fact that this research did no encompass assessment regarding players' learning gains or similar aspects. Since our testbed is a DMG, evaluating its benefits would be interesting in order to provide concerns about whether it can aid players on learning or improving their math knowledge. However, the key idea of the game is not to teach the four basic arithmetic operations, rather, it aims at fostering the practice of them. From this perspective, our analysis of whether the game made players curious about the subject tackled this aspect. It captured if the game raised questions about the subject (C6), if it was played because players wanted to know more about it (C7) or even if it motivated them to learn more about math (C1). Hence, the main educational aspect of the game, that is to make players practice math, was evaluated in terms of whether it aroused players curiosity. Nevertheless, future researches could go further on similar analyses, implementing a pre and post-test intervention to provide empirical evidence about how it influences players' learning or using an evaluation metric such as the Flow [86], which has demonstrated to increase students' learning [87].

7 FINAL CONSIDERATIONS

PCG is a valuable technique to improve game development, however, its influences on PX has been scantly investigated. This work presented an analysis of which influences the use of *generic* PCG arouses to players in comparison to human-authored levels. To this end, data regarding 724 players were assessed in terms of their opinions and in-game behavior, comparing their experience in each game version.

We found that both game versions provide players with positive experiences. The procedurally generated content was able to promote experiences as good as the human-designed levels in all but one factor, where this single difference mainly emerges by demographic characteristics. However, in terms of behavior, the *dynamic* version retained and engaged players as much as the *static* one. On the other hand, concerning players performance, the *static* version was less challenger than the *dynamic* one in terms of average score and game progression, which might also have affected the PX difference. Despite that, we provided evidence that the employed PCG method can have its contents' difficulty controlled through the generation parameter and remedy this discrepancy.

Furthermore, we found that players' experienced fun and willingness to play the game again have strong highly significant correlations to their curiosity. Then, providing evidence that improving their experience is likely to also improve their math curiosity, besides the curiosity about the game itself. Moreover, the experiments showed that both players' demographics and in-game performance have a small to moderate impact on their experiences. Thus, this research's main goal was achieved by showcasing the influences of PCG on players from a DMG, whereas evaluating the learning aspects that this approach might provide was out of its scope.

In sum, our main findings are: (1) developers can benefit from the advantages of PCG to development while providing experiences equivalent to human-designed content in terms of behavior and nearly all self-reported factors and, yet control the content's difficulty to promote similar challenging levels; (2) our testbed game can be used to improve players' interest by means of always providing new contents, at the same time that fosters the practice of math; (3) players fun and *returnance* have strong impacts on their curiosity; and (4) both demographic and in-game data affect PX with a degree of correlation that ranges from small to moderate.

7.1 Future Works

Based on the discussion of this research's limitations and future directions to address them, we have drawn the following lines for future works in similar researches:

- To use more than one game version with human-designed content;
- To employ different techniques to procedurally generate levels;
- To investigate PCG's influences based on different game genres;
- To provide the SpaceMath game as a testbed to compare other PCG techniques to the same baseline;
- To adopt players' learning gains as evaluation metrics;
- To apply machine learning techniques to develop player models based on both demographic and in-game data;
- To develop an adaptive game version, based on the experience-driven PCG perspective, and assess its influences through the same methodology.

REFERENCES

[1] MEC. *Parâmetros Curriculares Nacionais: Matemática*. 1997. In portuguese. Disponível em: <portal.mec.gov.br/seb/arquivos/pdf/livro03.pdf>.

[2] NCMST. *Before It's Too Late: A Report to the Nation from the National Commission on Mathematics and Science Teaching for the 21st Century*. [S.l.], 2000.

[3] OCDE. *PISA 2015 Results in Focus*. 2015. Access in: July 16, 2018. Disponível em: <oecd.org/pisa/pisa-2015-results-in-focus.pdf>.

[4] BISWAS, G. et al. Extending intelligent learning environments with teachable agents to enhance learning. p. 389–397, 10 2001.

[5] MADEIRA, C. et al. Mathmare: um jogo de plataforma envolvendo desafios matemáticos do ensino médio. In: *Proceedings of the Brazilian Symposium on Computer Games and Digital Entertainment (SBGames 2015)*. [S.l.: s.n.], 2015. In portuguese.

[6] MCLAREN, B. M. et al. A computer-based game that promotes mathematics learning more than a conventional approach. *International Journal of Game-Based Learning (IJGBL)*, v. 7, p. 36–56, 2017.

[7] BRAZIL, A.; BARUQUE, L. Gamificação aplicada na graduação em jogos digitais. In: *Proceedings of the Brazilian Symposium on Computers in Education (SBIE 2015)*. [S.l.: s.n.], 2015. In portuguese.

[8] KE, F. A case study of computer gaming for math: Engaged learning from gameplay? *Computers & Education*, v. 51, n. 4, p. 1609 – 1620, 2008. ISSN 0360-1315. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0360131508000523>.

[9] KIILI, K.; KETAMO, H. Evaluating cognitive and affective outcomes of a digital game-based math test. *IEEE Transactions on Learning Technologies*, PP, n. 99, p. 1–1, 2017. ISSN 1939-1382.

[10] YURDABAKAN, I.; UZUNKAVAK, C. Primary school students' attitudes towards computer based testing and assessment in turkey. v. 13, p. 177–188, 07 2012.

[11] CARVALHO, M. F. de; GASPARINI, I.; HOUNSELL, M. da S. Digital games for math literacy: A systematic literature mapping on brazilian publications. In: ROCHA, Á. et al. (Ed.). *New Advances in Information Systems and Technologies*. Cham: Springer International Publishing, 2016. p. 245–254. ISBN 978-3-319-31307-8.

[12] KIILI, K.; MOELLER, K.; NINAUS, M. Evaluating the effectiveness of a game-based rational number training - in-game metrics as learning indicators. *Computers & Education*, v. 120, p. 13 – 28, 2018. ISSN 0360-1315. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0360131518300125>.

[13] IBARRA, M. J. et al. Mathfraction: Educational serious game for students motivation for math learning. In: *2016 XI Latin American Conference on Learning Objects and Technology (LACLO)*. [S.l.: s.n.], 2016. p. 1–9.

[14] CHENG, H. N. H. et al. Math detective: Digital game-based mathematical error dectection, correction and explanation. In: *2015 IEEE 15th International Conference on Advanced Learning Technologies*. [S.l.: s.n.], 2015. p. 122–126. ISSN 2161-3761.

[15] ARANTES, H.; SEABRA, R. Tme: Aplicativo m-learning para o estudo de conceitos matemáticos com ênfase no enem. In: *Proceedings of the Brazilian Symposium on Computers in Education (SBIE 2016)*. [S.l.: s.n.], 2016. In portuguese.

[16] ARAúJO, J. P. P. de; COSTA, G.; JúNIOR, J. G. R. Matematech: Plataforma de apoio à aprendizagem de matemática nos anos iniciais do ensino fundamental. In: *Proceedings of the Brazilian Symposium on Computers in Education (SBIE 2016)*. [S.l.: s.n.], 2016. In portuguese.

[17] HENDRIKX, M. et al. Procedural content generation for games: A survey. *ACM Trans. Multimedia Comput. Commun. Appl.*, ACM, New York, NY, USA, v. 9, n. 1, p. 1:1–1:22, feb 2013. ISSN 1551-6857.

[18] AMATO, F.; MOSCATO, F. Formal procedural content generation in games driven by social analyses. In: *2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA)*. [S.l.: s.n.], 2017. p. 674–679.

[19] TOGELIUS, J. et al. What is procedural content generation?: Mario on the borderline. In: *Proceedings of the 2nd International Workshop on Procedural Content Generation in Games (PCGames '11)*. [S.l.: s.n.], 2011.

[20] CARLI, D. et al. A survey of procedural content generation techniques suitable to game development. In: *Brazilian Symposium on Computer Games and Digital Entertainment (SBGAMES 2011)*. [S.l.: s.n.], 2011. p. 26–35. ISSN 2159-6654.

[21] HORN, B. et al. A comparative evaluation of procedural level generators in the mario ai framework. In: *Foundations of Digital Games 2014*. [s.n.], 2014. Disponível em: <http://www.fdg2014.org/>.

[22] KORN, O. et al. Procedural content generation for game props? a study on the effects on user experience. *Comput. Entertain.*, ACM, New York, NY, USA, v. 15, n. 2, p. 1:1–1:15, abr. 2017. ISSN 1544-3574. Disponível em: <http://doi.acm.org/10.1145/2974026>.

[23] SMITH, G.; WHITEHEAD, J. Analyzing the expressive range of a level generator. In: . New York, NY, USA: ACM, 2010. (Proceedings of the 1st International Workshop on Procedural Content Generation in Games (PCGames '10)), p. 4:1–4:7. ISBN 978-1-4503-0023-0. Disponível em: <http://doi.acm.org/10.1145/1814256.1814260>.

[24] MOGHADAM, A. B.; RAFSANJANI, M. K. A genetic approach in procedural content generation for platformer games level creation. In: *2017 2nd Conference on Swarm Intelligence and Evolutionary Computation (CSIEC)*. [S.l.: s.n.], 2017. p. 141–146.

[25] HORN, B. et al. Design insights into the creation and evaluation of a computer science educational game. In: *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*. New York, NY, USA: ACM, 2016. (SIGCSE '16), p. 576–581. ISBN 978-1-4503-3685-7. Disponível em: <http://doi.acm.org/10.1145/2839509.2844656>.

[26] ARAúJO, W.; ARANHA, E. Geração procedural de conteúdo para criação de fases de jogos educativos usando gramática. In: *Proceedings of the Brazilian Symposium on Computers in Education*. [S.l.: s.n.], 2018. In Portuguese.

[27] DONG, Y.; BARNES, T. Evaluation of a template-based puzzle generator for an educational programming game. In: *Proceedings of the 12th International Conference on the Foundations of Digital Games*. New York, NY, USA: ACM, 2017. (FDG '17), p. 40:1–40:4. ISBN 978-1-4503-5319-9. Disponível em: <http://doi.acm.org/10.1145/3102071.3106347>.

[28] RODRIGUES, L.; BONIDIA, R. P.; BRANCHER, J. D. A math educational computer game using procedural content generation. In: *Proceedings of the Brazilian Symposium on Computers in Education (SBIE 2017)*. [s.n.], 2017. Disponível em: <http://www.brie.org/pub/index.php/sbie/article/view/7604/5400>.

[29] BAUCKHAGE, C. et al. How players lose interest in playing a game: An empirical study based on distributions of total playing times. In: *2012 IEEE Conference on Computational Intelligence and Games (CIG)*. [S.l.: s.n.], 2012. p. 139–146. ISSN 2325-4270.

[30] SIM, G.; HORTON, M. Investigating children's opinions of games: Fun toolkit vs. this or that. In: *Proceedings of the 11th International Conference on Interaction Design and Children*. New York, NY, USA: ACM, 2012. (IDC '12), p. 70–77. ISBN 978-1-4503-1007-9. Disponível em: <http://doi.acm.org/10.1145/2307096.2307105>.

[31] RODRIGUES, L.; BRANCHER, J. D. Improving players' profiles clustering from game data through feature extraction. In: *Proceedings of SBGames 2018 - Computing Track*. [s.n.], 2018. Disponível em: <sbgames.org/sbgames2018/files/papers/ComputacaoFull/188130.pdf>.

[32] SAMPAYO-VARGAS, S. et al. The effectiveness of adaptive difficulty adjustments on students' motivation and learning in an educational computer game. *Computers & Education*, v. 69, p. 452 – 462, 2013. ISSN 0360-1315. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0360131513001711>.

[33] OLIVEIRA, S.; MAGALHãES, L. Adaptive content generation for games. In: *2017 24º Encontro Português de Computação Gráfica e Interação (EPCGI)*. [S.l.: s.n.], 2017. p. 1–8.

[34] PAPADIMITRIOU, S.; VIRVOU, M. Adaptivity in scenarios in an educational adventure game. In: *2017 8th International Conference on Information, Intelligence, Systems Applications (IISA)*. [S.l.: s.n.], 2017. p. 1–6.

[35] TOGELIUS, J. et al. Search-based procedural content generation: A taxonomy and survey. *Computational Intelligence and AI in Games, IEEE Transactions on*, v. 3, n. 3, p. 172–186, Sept 2011. ISSN 1943-068X.

[36] LOPES, R.; BIDARRA, R. Adaptivity challenges in games and simulations: A survey. *IEEE Transactions on Computational Intelligence and AI in Games*, v. 3, n. 2, p. 85–99, June 2011. ISSN 1943-068X.

[37] HENDRIX, M. et al. Implementing adaptive game difficulty balancing in serious games. *IEEE Transactions on Games*, p. 1–9, 2018. ISSN 2475-1502.

[38] KARPINSKYJ, S.; ZAMBETTA, F.; CAVEDON, L. Video game personalisation techniques: A comprehensive survey. *Entertainment Computing*, v. 5, n. 4, p. 211 – 218, 2014. ISSN 1875-9521. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1875952114000342>.

[39] SILVA, M. P.; SILVA, V. d. N.; CHAIMOWICZ, L. Dynamic difficulty adjustment through an adaptive ai. In: *2015 14th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*. [S.l.: s.n.], 2015. p. 173–182.

[40] YANNAKAKIS, G. N.; TOGELIUS, J. Experience-driven procedural content generation. *IEEE Transactions on Affective Computing*, v. 2, n. 3, p. 147–161, July 2011. ISSN 1949-3045.

[41] OLIVEIRA, W.; SILVA, T. R.; ARANHA, E. Aplicação de jogos adaptativos na educação: uma revisão sistemática da literatura. In: *Proceedings of the Brazilian Symposium on Computers in Education (SBIE 2016)*. [S.l.: s.n.], 2016.

[42] YANNAKAKIS P. SPRONCK, D. L. G. N.; ANDRE, E. Player modeling. In: AL., J. T. et (Ed.). [S.l.]: Dagstuhl Seminar on Artificial and Computational Intelligence in Games, 2013.

[43] CONNOR, A. M.; GREIG, T. J.; KRUSE, J. Evaluating the impact of procedurally generated content on game immersion. *The Computer Games Journal*, v. 6, n. 4, p. 209–225, Dec 2017. ISSN 2052-773X. Disponível em: <https://doi.org/10.1007/s40869-017-0043-6>.

[44] BUTLER, E. et al. Automatic game progression design through analysis of solution features. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. New York, NY, USA: ACM, 2015. (CHI '15), p. 2407–2416. ISBN 978-1-4503-3145-6. Disponível em: <http://doi.acm.org/10.1145/2702123.2702330>.

[45] TOGELIUS, J.; JUSTINUSSEN, T.; HARTZEN, A. Compositional procedural content generation. In: *Proceedings of the The Third Workshop on PCG in Games*. New York, NY, USA: ACM, 2012. (PCG'12), p. 16:1–16:4. ISBN 978-1-4503-1447-3. Disponível em: <http://doi.acm.org/10.1145/2538528.2538541>.

[46] CHEONG, Y. G.; YOUNG, R. M. Suspenser: A story generation system for suspense. *IEEE Transactions on Computational Intelligence and AI in Games*, v. 7, n. 1, p. 39–52, March 2015. ISSN 1943-068X.

[47] WOLOSZYN, V. et al. Beatnik: an algorithm to automatic generation of educational description of movies. In: *Proceedings of the Brazilian Symposium on Computers in Education (SBIE 2017)*. [S.l.: s.n.], 2017.

[48] SILVEIRA, I. et al. Real-time procedural generation of personalized facade and interior appearances based on semantics. In: *2015 14th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*. [S.l.: s.n.], 2015. p. 89–98.

[49] RODRíGUEZ, J. C.; DíAZ, R. A. F.; CARRIEGOS, M. V. Automatic generation of moodle questionnaires to assess learning of descriptive geometry. In: *2014 International Symposium on Computers in Education (SIIE)*. [S.l.: s.n.], 2014. p. 201–204.

[50] ETCHEBEHERE, G. S.; ELISEO, M. A. L-systems and procedural generation of virtual game maze sceneries. In: *Proceedings of the Brazilian Symposium on Computer Games and Digital Entertainment (SBGames 2017)*. [S.l.: s.n.], 2017.

[51] SMITH, A.; MATEAS, M. Answer set programming for procedural content generation: A design space approach. *Computational Intelligence and AI in Games, IEEE Transactions on*, v. 3, n. 3, p. 187–200, Sept 2011. ISSN 1943-068X.

[52] van der Linden, R.; LOPES, R.; BIDARRA, R. Designing procedurally generated levels. In: _____. *Proceedings of IDPv2 2013 - Workshop on Artificial Intelligence in the Game Design Process, Ninth AAAI Conference, on Artificial Intelligence in Interactive Digital Entertainment*. United States: American Association for Artificial Intelligence (AAAI), 2013. p. 41–47. ISBN 978-1-57735-635-6.

[53] DORAN, J.; PARBERRY, I. Controlled procedural terrain generation using software agents. *Computational Intelligence and AI in Games, IEEE Transactions on*, v. 2, n. 2, p. 111–119, June 2010. ISSN 1943-068X.

[54] VISUALIZATION, I. I. D. *SpeedTree*. 2017. Disponível em: <http://www.speedtree.com/>.

[55] CARDAMONE, L.; LOIACONO, D.; LANZI, P. L. Interactive evolution for the procedural generation of tracks in a high-end racing game. In: *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*. New York, NY, USA: ACM, 2011. (GECCO '11), p. 395–402. ISBN 978-1-4503-0557-0. Disponível em: <http://doi.acm.org/10.1145/2001576.2001631>.

[56] KARAVOLOS, D.; BOUWER, A.; BIDARRA, R. Mixed-initiative design of game levels: integrating mission and space into level generation. In: *Foundations of Digital Games 2015*. [s.n.], 2015. Disponível em: <http://graphics.tudelft.nl/Publications-new/2015/KBB15>.

[57] SCIREA, M. et al. Evaluating musical foreshadowing of videogame narrative experiences. In: *Proceedings of the 9th Audio Mostly: A Conference on Interaction With Sound*. New York, NY, USA: ACM, 2014. (AM '14), p. 8:1–8:7. ISBN 978-1-4503-3032-9. Disponível em: <http://doi.acm.org/10.1145/2636879.2636889>.

[58] FERREIRA, L.; PEREIRA, L.; TOLEDO, C. A multi-population genetic algorithm for procedural generation of levels for platform games. In: *Proceedings of the 2014 Conference Companion on Genetic and Evolutionary Computation Companion*. New York, NY, USA: ACM, 2014. (GECCO Comp '14), p. 45–46. ISBN 978-1-4503-2881-4. Disponível em: <http://doi.acm.org/10.1145/2598394.2598489>.

[59] SHAKER, N. et al. Evolving personalized content for super mario bros using grammatical evolution. In: *Proceedings of the Eighth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. AAAI Press, 2012. (AIIDE'12), p. 75–80. Disponível em: <http://dl.acm.org/citation.cfm?id=3014629.3014643>.

[60] LUO, L. et al. Design and evaluation of a data-driven scenario generation framework for game-based training. *IEEE Transactions on Computational Intelligence and AI in Games*, v. 9, n. 3, p. 213–226, Sept 2017. ISSN 1943-068X.

[61] MACHADO, M. C.; FANTINI, E. P. C.; CHAIMOWICZ, L. Player modeling: Towards a common taxonomy. In: *2011 16th International Conference on Computer Games (CGAMES)*. [S.l.: s.n.], 2011. p. 50–57.

[62] CARNEIRO, E. M.; CUNHA, A. M. d.; DIAS, L. A. V. Adaptive game ai architecture with player modeling. In: *2014 11th International Conference on Information Technology: New Generations*. [S.l.: s.n.], 2014. p. 40–45.

[63] MARIÑO, J. R. H.; REIS, W. M. P.; LELIS, L. H. S. An empirical evaluation of evaluation metrics of procedurally generated mario levels. In: *Proceedings of the Eleventh Artificial Intelligence and Interactive Digital Entertainment*. [S.l.: s.n.], 2015. p. 44–50.

[64] RODRIGUES, L. A. L. et al. *SpaceMath: Praticando Matemática no Espaço*. 2018. Disponível em: <spacemath.rpbtecnologia.com.br>.

[65] KHALIFA, A. et al. General video game level generation. In: *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2016)*. [S.l.: s.n.], 2016. p. 253–259.

[66] READ, J.; MACFARLANE, S.; CASEY, C. Endurability, engagement and expectations: Measuring children's fun. 01 2009.

[67] MOSER, C.; FUCHSBERGER, V.; TSCHELIGI, M. Rapid assessment of game experiences in public settings. In: *Proceedings of the 4th International Conference on Fun and Games*. New York, NY, USA: ACM, 2012. (FnG '12), p. 73–82. ISBN 978-1-4503-1570-8. Disponível em: <http://doi.acm.org/10.1145/2367616.2367625>.

[68] WOUTERS, P. et al. The role of game discourse analysis and curiosity in creating engaging and effective serious games by implementing a back story and foreshadowing. *Interacting with Computers*, v. 23, n. 4, p. 329 – 336, 2011. ISSN 0953-5438. Cognitive Ergonomics for Situated Human-Automation Collaboration. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0953543811000415>.

[69] NPD. *New Report from The NPD Group Provides In-Depth View of Brazil's Gaming Population*. 2015. Disponível em: <www.npd.com/wps/portal/npd/us/news/press-releases/2015/new-report-from-the-npd-group-provides-in-depth-view-of-brazils-gaming-population/>.

[70] R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2018. Disponível em: <https://www.R-project.org/>.

[71] MANN, H. B.; WHITNEY, D. R. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, JSTOR, p. 50–60, 1947.

[72] KRUSKAL, W. H.; WALLIS, W. A. Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, Taylor & Francis Group, v. 47, n. 260, p. 583–621, 1952.

[73] RAO, J. N.; SCOTT, A. J. The analysis of categorical data from complex sample surveys: chi-squared tests for goodness of fit and independence in two-way tables. *Journal of the American statistical association*, Taylor & Francis, v. 76, n. 374, p. 221–230, 1981.

[74] SHAKER, N.; YANNAKAKIS, G.; TOGELIUS, J. Towards automatic personalized content generation for platform games. In: *6th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AIIDE 2010*. [S.l.: s.n.], 2010.

[75] PEDERSEN, C.; TOGELIUS, J.; YANNAKAKIS, G. N. Modeling player experience for content creation. *IEEE Transactions on Computational Intelligence and AI in Games*, v. 2, n. 1, p. 54–67, March 2010. ISSN 1943-068X.

[76] WEBER, B. G. et al. Modeling player retention in madden nfl 11. In: AAAI PRESS. *Innovative Applications of Artificial Intelligence (IAAI)*. San Francisco, CA: AAAI Press, 2011.

[77] MAHLMANN, T. et al. Predicting player behavior in tomb raider: Underworld. In: *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*. [S.l.: s.n.], 2010. p. 178–185. ISSN 2325-4270.

[78] DRACHEN, A. et al. Rapid Prediction of Player Retention in Free-to-Play Mobile Games. In: *Proceedings of AAAI Artificial Intelligence and Interactive Digital Entertainment*. [S.l.: s.n.], 2016.

[79] KENDALL, M. G. A new measure of rank correlation. *Biometrika*, JSTOR, v. 30, n. 1/2, p. 81–93, 1938.

[80] STATISTICS, L. *Pearson's Product-Moment Correlation using SPSS Statistics*. 2018. Disponível em: <https://statistics.laerd.com/spss-tutorials/pearsons-product-moment-correlation-using-spss-statistics.php>.

[81] STATISTICS, L. *Kendall's Tau-b using SPSS Statistics*. 2018. Disponível em: <https://statistics.laerd.com/spss-tutorials/kendalls-tau-b-using-spss-statistics.php>.

[82] WALKER, D. A. Jmasm9: Converting kendall's tau for correlational or meta-analytic analyses. v. 2, p. 525–530, 11 2003.

[83] COHEN, J. Statistical power analysis for the behavioral sciences. v. 2nd, 1988.

[84] SCIREA, M. et al. Evolving in-game mood-expressive music with metacompose. In: *Proceedings of Audio Mostly (AM 2018)*. [S.l.: s.n.], 2018.

[85] BERKHIN, P. A survey of clustering data mining techniques. In: _____. *Grouping Multidimensional Data: Recent Advances in Clustering*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006. p. 25–71. ISBN 978-3-540-28349-2. Disponível em: <https://doi.org/10.1007/3-540-28349-8_2>.

[86] CSIKSZENTMIHALYI, M. *Flow: The Psychology of Optimal Experience*. [S.l.]: HarperCollins, 2009. (Harper Perennial Modern Classics). ISBN 9780061876721.

[87] SANTOS, W. O. dos et al. Flow theory to promote learning in educational systems: Is it really relevant? *BRAZILIAN JOURNAL OF COMPUTERS IN EDUCATION*, v. 26, n. 02, p. 29–59, 2018.

PUBLICATIONS

Works published by the author during the Master's Degree:

Main publications

1. Luiz A. L. Rodrigues, Jacques D. Brancher, **Improving Players' Profiles Clustering from Game Data Through Feature Extraction**, Proceedings of the XVII Brazilian Symposium on Computer Games and Digital Entertainment (SBGames), 10/2018, SBC, p. 1-10, ISSN 2179-2259, (Qualis CC, B2)

2. Luiz A. L. Rodrigues, Robson P. Bonidia, Jacques D. Brancher, **A Math Educational Computer Game Using Procedural Content Generation**, Proceedings of the XXVIII Brazilian Symposium on Computers in Education (SBIE), 10/2017, SBC, p. 756-765, ISSN 2316-6533, (Qualis CC, B1)

Complementary publications

1. Robson P. Bonidia, Luiz A. L. Rodrigues, Anderson P. Avila-Santos, Danilo S. Sanches, Jacques D. Brancher, **Computational Intelligence in Sports: A Systematic Literature Review**, Advances in Human-Computer Interaction, Hindawi, 2018, p. 1-13, ISSN 1687-5893, (Qualis CC, B2)

2. Renato Kuroe, Luiz A. L. Rodrigues, Robson P. Bonidia, Danilo Sanches, Jacques D. Brancher, Willyan Nazima, Fabrício Furtado, **EcoCloud: A Specialized Computer System for Elaboration of Echocardiography Reports**, Proceedings of the 24TH Americas Conference on Information Systems, 08/2018, AIS, p. 1-10, ISBN 978-0-9966831-6-6, (Qualis CC, A2)

3. Robson P. Bonidia, Luiz A. L. Rodrigues, Jacques D. Brancher, Roberto S. do Carmo, Carolina Massae, **Sistema Integrado de Gestão Esportiva: uma Ferramenta de Apoio ao Programa Talento Olímpico do Paraná**, Proceedings of the XIII Brazilian Symposium on Information Systems (SBSI), 06/2017, SBC, p. 143-150, ISBN 978-85-7669-376-5, (Qualis CC, B2)

4. Ronan A. Lopes, Luiz A. L. Rodrigues, Jacques D. Brancher, **Predicting Master's Applicants Performance Using KDD Techniques**, Proceedings of the 12th Iberian Conference on Information Systems and Technologies (CISTI), 06/2017, IEEE, p. 1-6, ISBN 978-9-8998-4347-9