Towards Evaluating a Procedural Content Orchestrator Gameplay Data to Differentiate User Profiles

Leonardo Tórtoro Pereira[1], Tyago Yuji Teoi[2], Claudio Fabiano Motta Toledo[2]

[1]Instituto de Geociências e Ciências Exatas
Universidade Estadual Paulista (UNESP) – Rio Claro – SP – Brasil

[2]Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo (USP) – São Carlos – SP – Brasil

leonardo.t.pereira@unesp.br, claudio@icmc.usp.br, yujiteoi@gmail.com

Abstract. *We tested a procedural content orchestration algorithm against 15 anonymous users, against 12 different dungeons, played 119 times in total. We used questionnaires to collect data regarding player profiles, and gameplay data to analyze if could identify profiles using them only. Using PCA and clustering techniques, we were able to identify the most important attributes one may collect from gameplay data to analyze and differentiate play-styles. We also identified that the dungeon's characteristics have a heavy influence on analyzing profiles through gameplay, and a more controlled environment may be needed to identify player profiles. More data and further analysis are needed to extract player profiles from gameplay data, but preliminary results show promise.*

1. Introduction

One of the significant challenges in the multibillion-dollar gaming industry is understanding the diverse types of players and designing games that appeal to a broad audience [Lora et al. 2016]. Another challenge is, regardless of the player's skill level, adjusting the game's difficulty for a challenging, but not too punishing, player experience [Cowley et al. 2008]. A technique called procedural content generation (PCG), i.e., content creation through algorithms, is a path to address these challenges. Specifically, an adaptive PCG can create content by analyzing the player's interactions with the game in real-time [Togelius et al. 2016].

However, for extracting and analyzing player data, an adaptive PCG system needs strategies to define player profiles, estimate skill levels, and identify player motivation factors. These strategies often elevate the game development process into a user-centered design approach [Loria and Marconi 2018, Melhart et al. 2019]. Accurate player modeling with adaptive PCG can create an engaging player experience, increasing playing time while keeping the player entertained and challenged.

However, gaps in the literature still need to be explored to get closer to an ideal adaptive PCG system. One of the issues relates to the lack of PCG algorithms for creating content with distinct creative facets (e.g., levels, visuals, rules) simultaneously [Liapis et al. 2019]. The other is the lack of real-time PCG algorithms that can adapt their content in real-time.

This paper presents preliminary results from the analysis of implicit and explicit data collected from a real-time procedural content orchestrator, able to generate dungeons, missions, and enemies for different player profiles, expanding the generator presented in [Pereira et al. 2022]. Our study aims to provide insights on user evaluation using gameplay data, specially towards real-time PCG algorithms that may use gameplay data only to provide tailored content for users and adapt to them.

2. Related Work

We tested a procedural content orchestration algorithm against 15 anonymous users, against 12 different dungeons, played 119 times in total. We used questionnaires to collect data regarding player profiles and gameplay data to analyze if we could identify profiles using them only. Using PCA and clustering techniques, we identified the most important attributes one may collect from gameplay data to analyze and differentiate play styles. We also identified that the dungeon's characteristics heavily influence analyzing profiles through gameplay, and a more controlled environment may be needed to identify player profiles. More data and further analysis are needed to extract player profiles from gameplay data, but preliminary results show promise.

In this study, we focus on analyzing if and how gameplay data may be adequately used to infer a player's preference and play style, comparing it against explicit feedback from users. These data, mentioned in Jameson's categories for collecting data for "Adaptive Interfaces and Agents", as *nonexplicit input*, include all actions a user performs with the system which does not have the explicit purpose of revealing the user's information to the system [Jacko 2012, p. 318]. Therefore, works related to collecting and analyzing gameplay metrics are presented.

Four works stand out in this area [Heijne and Bakkes 2017, Melhart et al. 2019, Loria and Marconi 2018, de Lima et al. 2021]. The first collects dozens of game metrics, separated into different gameplay blocks of the research: the village tutorial, combat against enemies, puzzle solving, and exploration. These metrics were used to determine correlations between the player's profile and questionnaire responses regarding their difficulties and preferred activities. No conclusive result was obtained, but the analysis pointed out that a set of metrics may accomplish the desired goal [Heijne and Bakkes 2017].

The second work uses data from the *Ubisoft Perceived Experience Questionnaire* (UPEQ) and several gameplay metrics to feed machine-learning models. 26 metrics were collected, including player level, days played, duration of gaming sessions, among others. Several playing styles were derived from these observations by employing profiling techniques based on the sequence of player activities within the game. These models were found to be more accurate than 65% in the worst case and almost 80% in the best case.[Melhart et al. 2019]. Unlike our study, they used *support vector machine* models, with data from a specific third-person shooter game named *Tom Clancy's The Division*.

Using metrics from a gamified application, Loria and Marconi create abstract player behaviors and compare them with profiles obtained through the *Hexad* questionnaire. Some behaviors involved competitiveness, with players paying attention to the scoreboard and noticing when they reach the top 10 on the board. The relationship between metrics and *Hexad* types is low, but abstract actions may be capable of crafting more believable characters [Loria and Marconi 2018].

The fourth work, authored by de Lima, Feijó, and Furtado, modifies the game's narrative following the player's in-game actions, including combating enemies, utilizing items, and engaging with non-player characters. The story arc adds an element of tension to the narrative, which changes the narrative dynamics and provokes game events according to the action. Comparing a game with its quest-adaptive system and a base system without it revealed that their work resembles a more defined narrative trajectory based on player data in the game [de Lima et al. 2021].

The book by Bernhaupt contains a set of generic and gender-focused metrics employed in slightly older research and outside the context of procedural generation or player profiling. Some numbers show how long a game lasts, when important things happen, and when the player interacts with things in the game [Bernhaupt 2015].

3. Methodology

This section presents the method used to collect data from the player. Subsection 3.1 presents the game prototype used to conduct the experiments, including a description of collected implicit gameplay metrics. Next, Subsection 3.2 describes how we defined player profiles from a pre-test questionnaire.

3.1. Top-down game

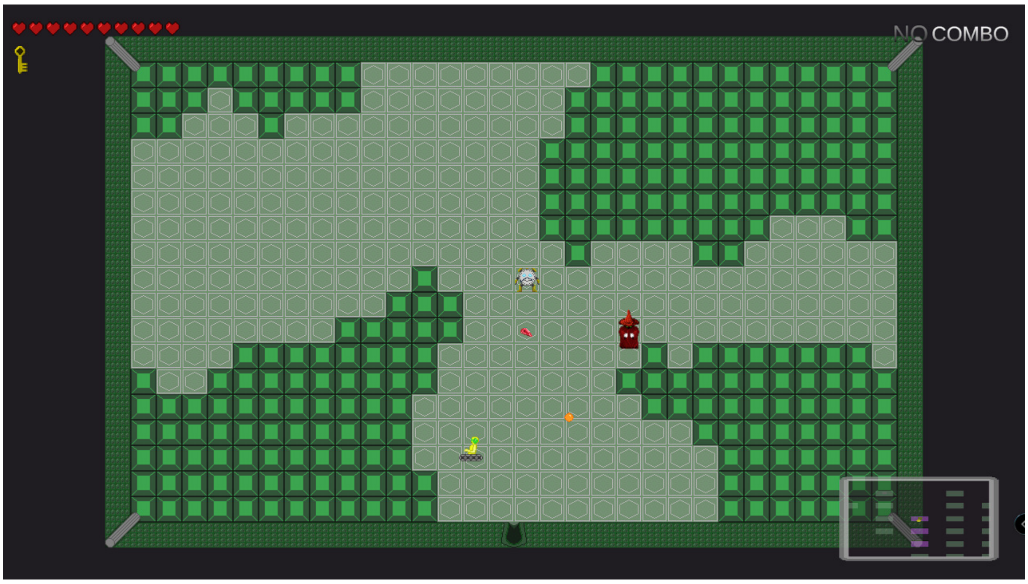

Figure 1. A screenshot of the game's dungeon. The player is the yellow character, the red mage is the enemy, the gray actor is a Non-player character, and the red gem is a pickable item.

The game was developed with *Unity*[1] game engine, using *C#* as a programming language. The game's project is free and open-source on *Github*[2].At the start of the game, it presents a pre-test questionnaire to the player (detailed in Subsection 3.2). After the player answers the questionnaire, the game screen changes to an overworld scene, where the player may choose between different dungeons, and start playing, as shown in Figure 1. Each player may play as many dungeons as they like, and we gather the gameplay data for each dungeon and user separately.

[1]https://unity.com/
[2]Removed for anonymity

The game is a Top-down genre game. The mechanics of the player and enemies consist of moving and shooting. Different enemies have different types of bullet projectiles or types of movement. Moreover, enemies with no shooting mechanic damage the player from touch. A room example is illustrated at the bottom of Figure 1. The yellow robot represents the player. A room can contain enemies, collectibles, doors, keys, and NPCs. Some doors are locked and require a key. NPCs ask the player to complete quests.

The winning condition is the player finding a *Triforce*, an interactive object inspired by *Legend of Zelda* instantiated in the dungeon. Therefore, completing quests, defeating enemies, exploring extra rooms, and collecting items are not mandatory. The loss condition is the player dying by taking enough damage to expire their life hearts.

Volunteers played the game through a link published on social networks and mailing lists. Before starting the game, they were shown a text explaining the research, which data we would collect and how. If they agreed, they would press the “Play” button and continue. A pre-test questionnaire was shown, so we could identify their profiles, and gameplay metrics were collected within Unity and sent to a private Firebase database as the player ends the game by winning or losing. The whole process was anonymous: anyone with the link could play, no personal data was collected, as names, IPs, and the like, and users were identified by a randomly generated number treated as their ID.

Table 1. Collected gameplay variables.

Variable	Data type	Description
Has finished	Boolean	Indicates whether the player found the Triforce or not.
Has died	Boolean	Indicates if the player died.
Lost health	Integer	Indicate how much health the player lost.
Max combo	Integer	Indicates the highest player’s combo. A combo is counted by the player hitting an enemy and is reseted wehn the player loses health.
Rooms visited	Integer	Indicates how many rooms (with repetition) were visited.
Unique rooms visited	Integer	Indicates how many unique rooms were visited.
Keys found	Integer	Indicates how many keys the player found.
Doors unlocked	Integer	Indicates how many doors the player unlocked.
Enemies defeated	Integer	Indicates how many enemies the player killed.
Completed quests	List of Integers	For each quest type, count how many were completed.

The collected gameplay variables are presented in Table 1.

The aforementioned quests were created with a similar algorithm as presented in [Pereira et al. 2022], with quests pertaining to each player type (*Achievement*, *Creativity*, *Immersion*, and *Mastery*). The first had quests of the *exchange* and *gather* archetypes, the second had quests for the *go to* and *explore* types, the third contained *give*, *listen*, *read*, and *report* quests, and the final type had only *kill* quests. All quests were extracted from the set proposed by [Doran and Parberry 2011]

3.2. Player profiling

The system’s player profiles are based on four motivation factors formalized by Yee et al. [Yee and Ducheneaut 2018]:

- Achievement: interest in power and completion, such as obtaining powerful items or completing all quests;

- Creativity: interest in design and discovery, such as character customization or exploring dungeons;
- Immersion: interest in fantasy and story, relating to the player's importance in the game world or rich character developments;
- Mastery: interest in challenge and strategy relating to difficult or strategy games.

We chose the given motivators because it is widely accepted in the industry and academic communities. Moreover, defined profiles using these motivators apply to many game genres, which is suitable to a goal for a PCG system to be usable for different game genres.

The used PCG system, adapted from [Pereira et al. 2022] defines the player's profile using a pre-test questionnaire with 12 questions. We built the questionnaire based on three-player modeling questionnaires, extracting crucial gameplay features for each player's motivator [Yee 2006, Rivera-Villicana et al. 2018, Vahlo et al. 2017]. Table 2 displays the used questionnaire.

Table 2. Pre-test questionnaire, presented to the player at the start of the game.

ID	Question
Q1	I am an experienced player.
Q2	I am an experienced player in the action-adventure genre.
Q3	In which difficulty do you usually play? (Options: Very easy, Easy, Medium, Hard, Very Hard)
Q4	I like playing games where I can explode, crush, destroy, shoot, and kill.
Q5	I like playing games where I can fight using close combat skills and evade fast attacks.
Q6	I like playing games where I can explore the game world and uncover secrets and mysteries.
Q7	I explore all the places, elements, and characters of the virtual world.
Q8	I complete all quests, including those that aren't necessary to finish the game.
Q9	I like playing games where I can collect rare items and hidden treasures.
Q10	I like playing games where I can build friendships between game characters and work toward a common goal.
Q11	I like playing games where I can immerse myself in the role of the character and make meaningful decisions.
Q12	I usually only do what is necessary to pass a level or complete a quest.

Except for Q3, the questions are presented on a 5-point Likert scale, with options ranging from Strongly Disagree (1) to Strongly Agree (5). From Equation 1, the weight of 7 is given to the highest-rated motivator, 5 for the second, 3 for the third, and 1 for the lowest-rated motivator. The content adaptation process using the player weights is the same as presented by [Pereira et al. 2022].

$$A = Q8 + Q9 + 5 - Q12 \qquad C = Q6 + Q7 + 5 - Q12 \tag{1}$$
$$I = Q10 + Q11 + 5 - Q12 \qquad M = Q3 - 3 + Q4 + Q5 \tag{2}$$

4. Results

Our game prototype was played by users with 15 different IDs (as our data acquisition was anonymous, the same user may have played more than once, and we would not be able to identify it), and their pre-test answers showed these users pertained to 8 different profiles. We collected data from 119 playthroughs from different dungeons (from a total of dungeons generated using 12 different input profiles, as seen in the markers in Figure 2, presented later). The profiles found in the pre-test follow: $\{A{=}4,\ C{=}3,\ I{=}2,\ M{=}1\}$;

{*A=2, C=1, I=3, M=4*}; {*A=3, C=2, I=4, M=1*}; {*A=4, C=2, I=1, M=3*}; {*A=1, C=3, I=2, M=4*}; {*A=2, C=1, I=4, M=3*}; {*A=3, C=2, I=1, M=4*}; {*A=4, C=1, I=3, M=2*}.

We first scaled the data with a min-max scaler, so that data could be distributed in the 0-1 range. This was specially helpful for the max combo, which had no upper limit. The other data was scaled from 0 to 1 considering their natural bounds. The exception was the total visited rooms, which was first divided by the dungeon's total rooms (presenting the backtracking ratio), and then was scaled with the min-max scaler.

Then, we describe the gameplay data statistical summary, presented in Table 3. and cluster the gameplay data from these 119 playthroughs to identify if we find any signs of the given profile over the gameplay metrics.

Table 3. Descriptive statistics for each gameplay variable, after min-max scaling.

Data	Max Combo	Completed Report Quests	Completed Listen Quests	Completed Achievement Quests	Lost Health	Completed Explore Quests	Completed Exchange Quests
Mean	0.176	0.101	0.101	0.131	0.527	0.126	0.0857
Std. Dev.	0.232	0.249	0.188	0.222	0.375	0.2156	0.174
Min.	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Max.	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Data	**Completed Kill Quests**	**Completed Mastery Quests**	**Completed Give Quests**	**Time To Finish**	**Completed Creativity Quests**	**Completed Immersion Quests**	**Completed Go To Quests**
Mean	0.109	0.109	0.081	0.223	0.117	0.1397	0.078
Std. Dev.	0.199	0.199	0.194	0.190	0.196	0.234	0.170
Min.	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Max.	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Data	**Completed Gather Quests**	**Completed Read Quests**	**Lock Usage Rate**	**Key Collected Rate**	**Map Completion**	**Room Revisit Rate**	**Enemy Kill Rate**
Mean	0.125	0.067	0.628	0.774	0.694	0.235	0.580
Std. Dev.	0.228	0.183	0.414	0.323	0.358	0.187	0.390
Min.	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Max.	1.0	1.0	1.0	1.0	1.0	1.0	1.0

As we may observe, the data was scaled by the min-max value to the range of 0-1. We observe that the players' combo (total hits against enemies without receiving damage) was, on average, lower than the best players, with a relatively high standard deviation (0.232). Overall, the completed quests were also, on average, lower than for the most completionist players, and having a standard deviation close to 0.2 for most quests. The data with larger standard deviation were mostly the ones not related to quests. Lost health, lock usage, and enemy kill had a standard deviation close to 0.4, and key collection was close to 0.3. This may suggest that a clustering algorithm will benefit from these data the most, as they have a higher entropy.

To further understand the data's entropy, while also reducing the dimensionality for clustering, we apply the Principal Component Analysis (PCA) algorithm to the data, reducing it to two dimensions. Applying PCA over the dataset to reduce it to two dimensions gives us the first component having 57% of the explained variability, and the second one having 9% of it. Therefore, these 2 components explain 65% of the data variability. A third component would explain 7% of it, and as it reduces visibility on further visualizations, we will analyze clustering after reducing data to 2 components.

Table 4 presents, for each component (row), the contribution (Eigenvalues) of each variable. The largest Eigenvalues show the most important variables contributing to each component. Therefore, the first component (which has the greatest explained variability) is largely formed by the information of *Lock Usage*, *Enemy Kill Rate*, *Map Completion*, *Key Collected Rate*, and *Lost Health*. The one is composed majorly by the values of *Completed Immersion Quests*, *Completed Report Quests*, *Key Collected Rate*, *Map Completion*, *Lock Usage Rate*, *Completed Achievement Quests*, *Completed Gather Quests*, and *Completed Give Quests*.

Table 4. Each variable's contribution to each principal component of the PCA.

Component	Max Combo	Completed Report Quests	Completed Listen Quests	Completed Achievement Quests	Lost Health	Completed Explore Quests	Completed Exchange Quests
0	0.02	0.14	0.12	0.16	0.34	0.15	0.10
1	0.21	0.33	0.19	0.27	0.09	0.20	0.18
Component	**Completed Kill Quests**	**Completed Mastery Quests**	**Completed Give Quests**	**Time To Finish**	**Completed Creativity Quests**	**Completed Immersion Quests**	**Completed Go To Quests**
0	0.12	0.12	0.11	0.18	0.14	0.18	0.10
1	0.08	0.08	0.24	0.10	0.21	0.36	0.16
Component	**Completed Gather Quests**	**Completed Read Quests**	**Lock Usage Rate**	**Key Collected Rate**	**Map Completion**	**Room Revisit Rate**	**Enemy Kill Rate**
0	0.15	0.09	0.44	0.31	0.38	0.18	0.40
1	0.25	0.19	0.29	0.31	0.29	0.09	0.10

This means that information related to locks, keys, enemies, map completion, and health are the most important to identify players by their gameplay, while quest-related data, specially those for the achievement, immersion, and creativity types has a minor but significant contribution. The first group of data is related to exploration (linked to the creativity type) and mastery, but may also relate to the achievement one, considering players that may consider exploring the whole dungeon an achievement. While the second group is more related to the immersion type (the most influential variable), and achievement. This shows that data that may relate to all four explored player types have a significant role in the explained variability aspect of the PCA, and may be useful to discover player's preferences through gameplay data alone.

Now, we apply clustering algorithms to explore if the gameplay data may be grouped in such a way that represents the original player's provided profiles by the pre-test. The first algorithm used to cluster the data is the K-Means algorithm. As we want to find a possible correlation to the player profiles, all algorithms will be executed with parameters related to creating 8 clusters. Figure 2 shows the generated clusters. At the bottom, the scatter plot of each gameplay data, considering the two principal components of the PCA, and grouping each playthrough by the player's profile (color) and the level's profile (marker), that is, the input provided to the generator to create the level's quests, enemies, and layout. This specific data may help visualize if different dungeons' configuration impact over the gameplay of each player profile.

We may observe the clustering was unable to precisely divide players by profile. However, some interesting groups emerged. Clusters C3 and C7 are majorly composed by players of the yellow profile (A=4, C=1, I=3, M=2), and both contain mostly data

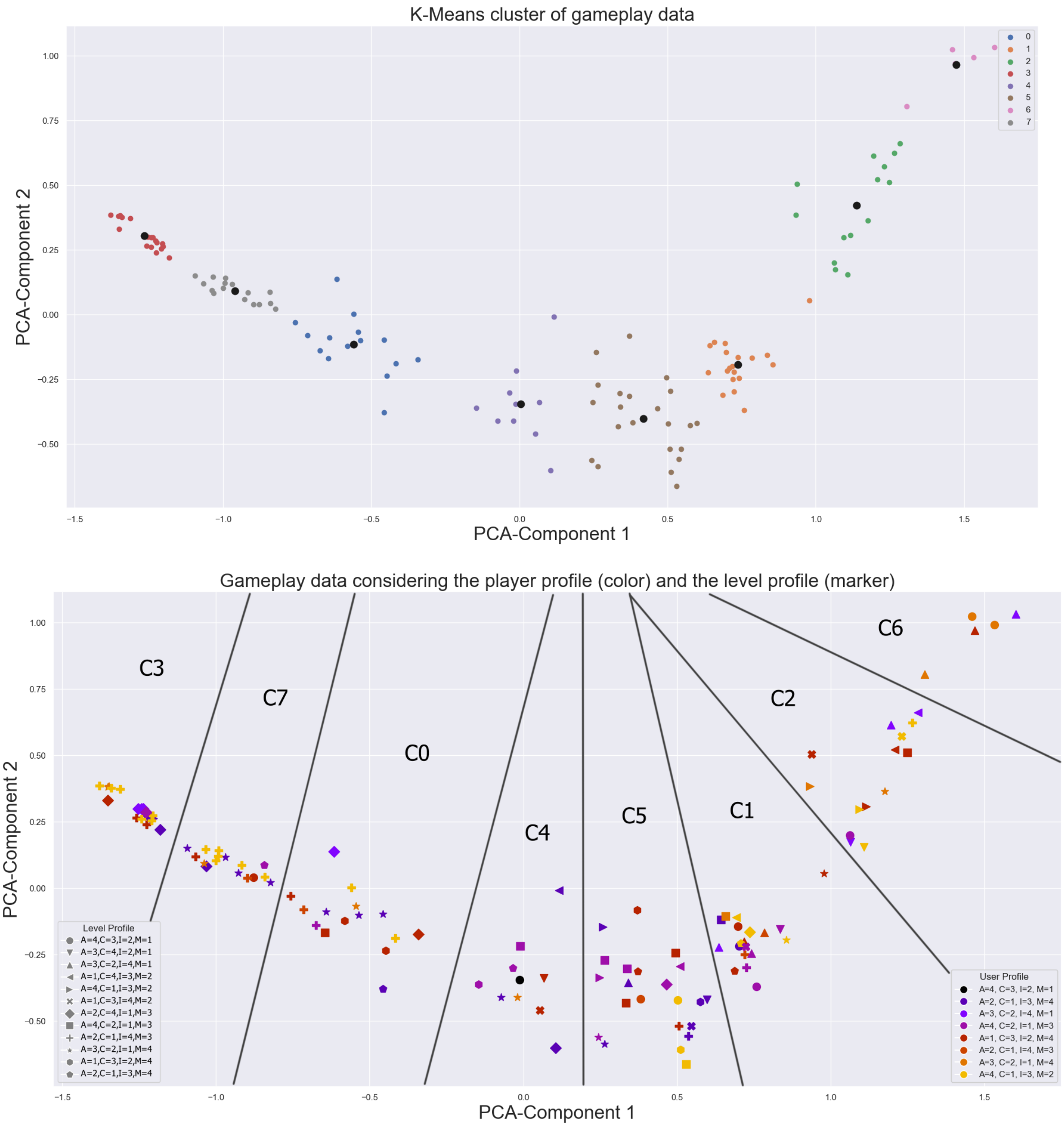

Figure 2. Scatter plot showing the generated clusters by the K-Means algorithm (top), and the gameplay data marked by player and level profile, with each cluster space marked by the lines.

from level *M=3, I=4, A=2, C=1*. C3 also contains the majority of data from level *M=3, I=1, A=4, C=2*, while C7 also holds data from *M=4, I=1, A=3, E=2*. This may be evidence that the level played has a significant impact on players, regardless of profile. Furthermore, said dungeons are focused on combat (high mastery), and are dungeons most players may have had more difficulty clearing them. As health and enemies killed were important variables for the variety in the PCA execution, the dungeon difficulty itself may impact more players than their profiles. We also observe cluster C0 has most data from the same dungeons as C7, but has players with a higher mastery profile (M=4 for most of them).

Cluster C4 and C5 have a high concentration of players with a high Mastery preference (M=3 and M=4), but ranging through various levels. C6 holds few data, but consists of mastery-liking players (M=4) that played low mastery levels (M=1). And C5 also contains players with high mastery preference, playing levels with a medium level of mastery (M=2 and M=3). For the other clusters (C1, C2, C4), they seem to hold a high variation of players and levels.

To further validate our analysis, a second clustering algorithm was used, the Spectral Clustering, also considering 8 clusters. It provided similar results, as shown in Figure 3. The Spectral's clusters C0, C1, C5, and C4 are almost identical to K-Means' clusters C1, C6, C2, and C3, respectively. Spectral's cluster C3 aggregates even more playthroughs of the *M=3, I=4, A=2, C=1* dungeon, from the same types of player than K-Means' cluster C0. Spectral's C2 holds more gameplay data from dungeon *M=4, I=1, A=3, C=2*, being a cluster of levels with difficult combat played by Mastery-oriented players, furthering the profile already perceived from K-Means' C0. Spectral's C6 and C7 are mostly a segregation of K-Means' C5, leaving easier dungeons in Spectral's C6, and the more difficult ones in C7.

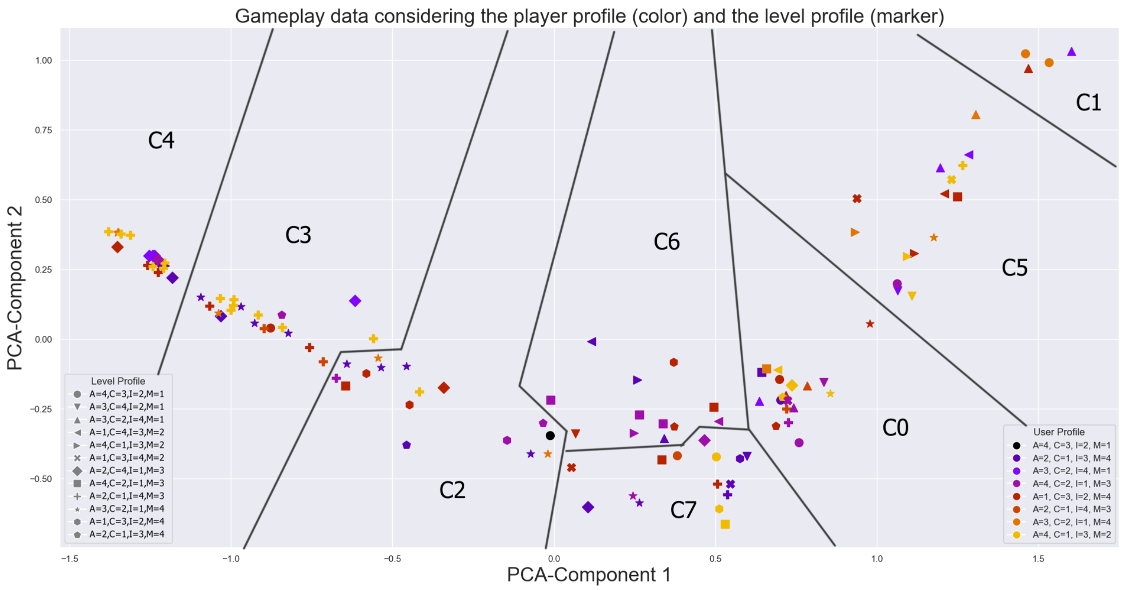

Figure 3. Scatter plot showing the gameplay data marked by player and level profile, with each cluster space marked by the lines, now generated by the Spectral Cluster algorithm (top).

5. Conclusion

We introduce a preliminary analysis on gameplay data from 119 playthroughs of 15 anonymous users, from 8 different profiles, playing through different dungeons aimed to target 12 different profiles, thanks to a procedural content orchestrator able to generate dungeons, quests, and enemies. Our focus was to find evidence in gameplay data that may help us identify a player's profile without the need for explicit input, such as questionnaires.

Through a PCA, we identified that data related to quest completion, time to finish a level, and how many hits a player gives without taking a hit were secondary to data related to elements such as enemies killed, health lost, keys collected, doors unlocked, and map completion. Signaling that these attributes are the most important to collect to evaluate users' gameplay behavior.

Then, we used clustering algorithms to identify if any behavior related to each user's profile would emerge. Although we could not trace from the cluster the player's profiles, we found that the level's components had a great impact on the clustering, specially how difficult their enemies were. Both clustering techniques used were mostly able to divide players and levels into categories related to: users who liked combat playing in dungeons with hard combat or easy combat, and users who disliked combat playing in dungeons with hard or easy combat.

Although more experiments are needed to draw better conclusions, our analysis was able to provide important insights. The first one is that the dungeon's difficulty may affect what is perceived as the user profile, and care must be taken in that regard. That is, a future study with more users playing the same dungeon (or a very reduced number of them) may provide data that allow clustering techniques to better identify player profiles, but generality may be hindered. The second one is that tracking which quests were completed by the player may not help to identify those who like immersion or achievements, as was expected. This may be caused by our test games lack of complex quests, but also may help future research to be careful when considering quest data.

Finally, the analysis provides insight that it is possible to group players using gameplay data, although we were not yet able to do so with the expected granularity, such as to define their specific player types and preferences. With more users, more data, and more implicit data collected, results may improve. However, these preliminary results may pave the way for real-time procedural content orchestration to use gameplay data to generate new content for users after each playthrough, creating a feedback loop to tailor content for each user's needs and improvement.

As possible future work, besides collecting more data, we plan to use classification algorithms to check their accuracy on classifying players with a given profile having only their gameplay data, and use the results in a real-time procedural content orchestration algorithm to analyze player's opinions on the quality of generated content and, therefore, the classification algorithm.

References

Bernhaupt, R. (2015). *Game user experience evaluation*. Springer.

Cowley, B., Charles, D., Black, M., and Hickey, R. (2008). Toward an understanding of flow in video games. *Computers in Entertainment (CIE)*, 6:1–27. DOI = `https://doi.org/10.1145/1371216.1371223`.

de Lima, E. S., Feijó, B., and Furtado, A. L. (2021). Adaptive branching quests based on automated planning and story arcs. In *2021 20th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, pages 9–18. DOI = `10.1109/SBGames54170.2021.00012`.

Doran, J. and Parberry, I. (2011). A prototype quest generator based on a structural analysis of quests from four mmorpgs. In *Proceedings of the 2nd International Workshop on Procedural Content Generation in Games*, PCGames '11, New York, NY, USA. Association for Computing Machinery.

Heijne, N. and Bakkes, S. (2017). Procedural zelda: a pcg environment for player experience research. In *Proceedings of the 12th International Conference on the Foundations of Digital Games*, FDG '17, New York, NY, USA. Association for Computing Machinery. DOI = `https://doi.org/10.1145/3102071.3102091`.

Jacko, J. A. (2012). *Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications, Third Edition*. CRC Press, Inc., Boca Raton, FL, USA, 3rd edition.

Liapis, A., Yannakakis, G. N., Nelson, M. J., Preuss, M., and Bidarra, R. (2019). Orchestrating game generation. *IEEE Transactions on Games*, 11(1):48–68.

Lora, D., Sánchez-Ruiz-Granados, A. A., González-Calero, P. A., and Gómez-Martín, M. A. (2016). Dynamic difficulty adjustment in tetris. In *FLAIRS Conference*. DOI = `10.1007/978-3-030-50353-6_1`.

Loria, E. and Marconi, A. (2018). Player types and player behaviors: Analyzing correlations in an on-the-field gamified system. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts, CHI PLAY 2018, Melbourne, VIC, Australia, October 28-31, 2018*, pages 531–538. DOI = `10.1145/3270316.3271526`.

Melhart, D., Azadvar, A., Canossa, A., Liapis, A., and Yannakakis, G. N. (2019). Your gameplay says it all: Modelling motivation in tom clancy's the division. *CoRR*, abs/1902.00040.

Pereira, L. T., Viana, B. M. F., and Toledo, C. F. M. (2022). A system for orchestrating multiple procedurally generated content for different player profiles. *IEEE Transactions on Games*, pages 1–11.

Rivera-Villicana, J., Zambetta, F., Harland, J., and Berry, M. (2018). Informing a bdi player model for an interactive narrative. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '18, page 417–428, New York, NY, USA. Association for Computing Machinery.

Togelius, J., Shaker, N., and Nelson, M. J. (2016). Introduction. In Shaker, N., Togelius, J., and Nelson, M. J., editors, *Procedural Content Generation in Games: A Textbook and an Overview of Current Research*, pages 1–15. Springer.

Vahlo, J., Kaakinen, J., Holm, S., and Koponen, A. (2017). Digital game dynamics preferences and player types: Preferences in game dynamics. *Journal of Computer-Mediated Communication*, 22.

Yee, N. (2006). Motivations for play in online games. *CyberPsychology & Behavior*, 9(6):772–775. PMID: 17201605.

Yee, N. and Ducheneaut, N. (2018). Gamer motivation profiling: Uses and applications. *Games User Research*, pages 485–490.