International Journal of Serious Games

ISSN: 2384-8766
https://journal.seriousgamessociety.org/

Article

Player Performance Estimation Through Adaptive Puzzle Levels Generation in Game-Based Assessment

Lailatul Husniah[1,2], M. Naufal Azzmi. H[3], Ali Sofyan Kholimi[2], Umi Laili Yuhana[4], Eko Mulyanto Yuniarno[1,5], and Mauridhi Hery Purnomo[1,5*]

[1]Departement of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia; [2]Departement of Informatics, Universitas Muhammadiyah Malang, Malang, Indonesia; [3]Department of Hospital Management Information System, West Nusa Tenggara Provincial General Hospital, Mataram, Indonesia; [4]Departement of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia; [5]Department of Computer Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

07111960010015@student.its.ac.id; naufalazzmi@gmail.com; kholimi@umm.ac.id; yuhana@if.its.ac.id; ekomulyanto@ee.its.ac.id; *hery@ee.its.ac.id

Keywords:

Game-based assessment
Player performance
Procedural content generation
Level generation
Puzzle game
Serious game

Received: May 2025
Accepted: October 2025
Published: October 2025
DOI: 10.17083/g3s30m71

Abstract

Estimating players' abilities without compromising alignment with assessment objectives remains a key challenge in developing Game-Based Assessment (GBA). This study proposes a method to estimate player performance through the procedural generation of puzzle game levels that adapt to player achievement. The method integrates Dynamic Difficulty Adjustment (DDA) and player modeling to dynamically align content difficulty with individual abilities while maintaining assessment validity. Unlike most previous PCG-GBA studies, which determine difficulty solely from technical parameters, the proposed method combines cognitive (arithmetic content) and technical (puzzle parameters) indicators, validated through paper-based tests. Implemented in a basic arithmetic puzzle game with sixth-grade primary school students in Indonesia, the approach generated 1,379 gameplay level records. Statistical analysis shows a strong positive correlation between paper-based test scores and the average difficulty of completed levels ($r = 0.734$, $p < 0.001$). A low absolute difference between difficulty level and player performance ($M = 0.012$, $SD = 0.028$) and a small RMSE (0.031) confirm the accuracy and consistency of difficulty adaptation. The proposed method enables accurate performance estimation through adaptive gameplay, contributing to the development and external validation of PCG-based GBA and pointing toward its potential scalability for personalized educational assessments.

1. Introduction

Over the past two decades, researchers, game designers, and educators in serious game research have faced the challenge of transforming games into practical and relevant assessment tools for educational contexts [1]. Assessment is at the core of the learning process because it provides essential feedback that identifies gaps in understanding, modifies learning strategies, monitors the achievement of learning outcomes, and measures the competency achievements of learners [2], [3]. Since assessment is at the core of the learning process and is very important for educators, developing game-based assessment (GBA) content should align with the stated assessment objectives [4]. Additionally, an efficient assessment process is crucial to enable periodic adjustments of difficulty levels based on student needs and performance [5], [6].

GBA offers a unique opportunity for alternative assessment and may reduce student stress compared to traditional testing methods [7], [8]. Students feel stressed during exams because they perceive exams as a threat to their self-esteem and abilities, which triggers anxiety when they feel unable to answer questions [9]. Test stress or anxiety can negatively impact student performance and lead to assessment bias [10]. Previous studies revealed that GBA can increase student engagement and reduce potential bias caused by test anxiety [10]. Moreover, GBA enables personalized feedback tailored to each student's challenges [11], [12]. Therefore, developing GBA content, especially personalized game-level design, is crucial. However, one of the main challenges in GBA is how to accommodate the diversity of students' abilities to provide problems or challenges that match their abilities [13]. Dynamic difficulty adjustment (DDA) is one approach to addressing these challenges. It has been proven effective in automatically adjusting the game difficulty based on individual player abilities [14]. In addition, a difficulty level control and verification mechanism is also needed so that the generated content remains relevant and aligned with the assessment objectives [11], [15].

Game levels are core elements of game content that determine the challenge level faced by the player [16], [17]. In the GBA context, automatically generated game levels with adaptive difficulty levels potentially improve assessment effectiveness by tailoring challenges to the player's abilities. To support this process, procedural content generation (PCG) comes as an algorithmic solution to generate varied, consistent, and controllable game content [16], [18], [19]. PCG has proven to be effective and widely used in entertainment game content development to generate various types of game content [17], [20], [21].

Although several studies have applied PCG in various contexts, such as game development based on design [22], [23], player preferences [14], [24], game difficulty estimation [25]–[27], to specific genres [28], [29], its application in the educational game domain is still relatively limited. Several studies have integrated PCG into the academic context to generate math problems for elementary education, utilizing textual and visual representations in a non-game context [30] and to create game content that improves English reading skills [31]. However, some PCG studies in the educational context focus on the development of GBL rather than GBA. GBL is generally oriented toward supporting the learning process, whereas GBA focuses more on assessing players' skills or knowledge based on their interactions with the game [12]. These limitations indicate the need for a new approach to automatically generate game content and support adaptive and relevant assessment of player abilities in an educational context.

Therefore, this study proposes an adaptive puzzle-level generation method for GBA by integrating DDA principles and player modeling approaches. The generation process is procedurally designed to adjust difficulty levels based on player achievements, ensuring that the resulting content remains valid, personalized, and aligned with assessment objectives. The method enables the estimation of player performance and the automatic adjustment of difficulty for subsequent levels, providing challenges that match individual abilities. This approach is expected to support the development of adaptive, personalized, and innovative GBA systems. This study aims to address the following research questions:

1) Question 1: How can assessment standards be transformed into variables that define the difficulty level of game levels?
2) Question 2: How can the GBA system procedurally generate game levels with difficulty levels that adapt to player performance?
3) Question 3: How does the GBA system estimate player performance based on their gameplay performance?

This paper organizes its content as follows: Section 2 reviews research on GBA and the use of PCG in the context of educational games. Section 3 describes the proposed adaptive game level generation method, including integrating difficulty level criteria and procedural content generation based on player performance data to estimate their abilities. Section 4 describes the experiment and evaluation design, including data collection from player interaction and performance analysis. Finally, in Section 5, we summarize our findings and suggest future research directions.

2. Related Work

2.1 Games-based assessment

GBA is an assessment approach that uses specially designed serious games to measure abilities (e.g., knowledge, skills) or specific attributes, such as motivation or social skills, through gameplay activities [7], [12]. GBA creates an interactive and engaging assessment experience for players by integrating game elements and mechanics. In addition, GBA offers shorter assessment times and high data quality and quantity. The testing environment created is also more motivating through features such as real-time feedback, level progress, and clear goals [32].

Several studies have developed GBA in various educational contexts, such as to assess student understanding of physics concepts [33], rational numbers [34], fractions [35], and geometry [7], and then assess cognitive and social functioning in children and adolescents with autism [36], and measure students' design thinking choices [37]. In the medical field, GBA has been used to assess the cognitive function of Alzheimer's patients through simulated daily activities [38] and to assess and train the arm mobility of stroke patients through a telerehabilitation system [13]. The main objectives of GBA research include evaluations, behavioral studies in games, assessments, frameworks, and proposals for game design, with evaluation being the most common category [12].

While research in GBA has continued to expand, several challenges remain, such as translating assessment models into engaging game design elements, integrating competency models with suitable game mechanics, and using data analytics techniques to provide personalized feedback and instant analysis. In addition, one of the main limitations in developing GBA content is that game designs are not always specially designed for assessment purposes. Many studies use games initially developed for other purposes (e.g., entertainment) and ignore the critical link between game design and the collection of evidence required for assessment [12]. Most studies develop and design GBA content manually and specifically for a particular purpose. Therefore, an automation approach through techniques such as PCG is a promising solution to overcome these limitations.

2.2 PCG for educational game

PCG is a popular technique for automatic and dynamic game content generation. In the context of educational games, PCG not only supports development content efficiency but also has the potential to improve player motivation, engagement, and experience through constantly changing challenges [39]. One of the implementations of PCG is MentalMath, which generates

diverse scenarios and math problems in an adventure game designed to help children practice math [40]. However, this approach adjusts the difficulty based only on the number of levels played without directly considering the player's performance.

Several studies have integrated PCG with adaptive mechanisms that consider individual player's abilities. For example, the educational game Refraction uses a simple heuristic-based level design automation tool that utilizes PCG to generate levels according to the player's ability while introducing mathematical concepts [41]. While this approach speeds up the solution-finding process, the use of simple heuristics risks producing non-optimal or less aesthetically pleasing solutions. Hoosyar et al. [31] also applied a heuristic-based approach, which optimized the order of learning content based on designer input and previous player performance. The approach gives designers significant control in determining the intensity of the learning objectives and organizing the content to match the player's abilities. Meanwhile, another study developed a data-driven PCG approach to reduce reliance on designer intuition in adaptively adjusting game content [42].

Earlier studies integrate adaptive elements to developing GBL. The main objective of their approach is to produce game content that can be tailored to individual player's abilities to improve learning effectiveness. Furthermore, while not explicitly mentioning the terms DDA or player modeling, the approaches used in their studies have adopted the principles of both concepts. DDA plays a role in dynamically adjusting the game difficulty to match the player's ability, while player modeling allows the system to understand the player's playing patterns and characteristics to support such adjustments [14].

While DDA and player modeling-based approaches have great potential in supporting the development of adaptive GBA, the applicability of these methods in the context of GBA still needs further research. Table 1 summarizes the different approaches discussed to strengthen the justification of this study's novelty compared to previous research. This research focuses on developing a GBA that uses PCG to generate adaptive game levels, dynamically adjust level difficulty based on player performance data, and integrate DDA and player modeling principles to support player performance estimation.

Table 1. Research gap

Research	Puzzle genre	Mathematical Topic	GBA	PCG	Integrated DDA & player modeling
Kiili [34]	Yes	Yes	Yes	No	No
Kiili [35]	No	Yes	Yes	No	No
Ruiperez-Valiente [7]	Yes	Yes	Yes	No	No
Rodrigues and Brancher [39]	Yes	Yes	No	Yes	No
Smith [41]	Yes	Yes	No	Yes	Yes
Rodrigues [40]	No	Yes	No	Yes	No
Hooshyar [31]	No	No	No	Yes	Yes
Hooshyar [42]	No	No	No	Yes	Yes
Proposed method	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

3. Methods and Material

This section provides an overview of the steps involved in estimating player performance through adaptive puzzle-level generation in GBA.

3.1 Participants and Data Collection

3.1.1 Participants

This study involved all Grade 6 students from one public primary school in Indonesia as participants. The school selection was random with no specific criteria regarding location or school characteristics. Participants in this study were typically developing students with no reported diagnosis of developmental or neurological disorders. The principal, acting in loco parentis, approved and permitted the research, and ensured that the teacher conducted the activities during class time under their supervision and without any objection from parents. Classroom teachers conveyed information about the study to students, and student participation was entirely voluntary without coercion. Students were free to choose to participate or withdraw at any time without any consequences. The ethics committee also approved this study, and students who participated gave verbal consent.

A total of 30 students agreed to participate in the study, with 22 students following the full protocol and included in the data analysis. The participants consisted of ten male and twelve female students with an average age of 11.64 years (SD = 0.49). According to the mathematics teacher, the participants' mathematics abilities varied, as reflected in their previous midterm exam scores. The results of the paper-based test in this study, which showed a mean score of 0.42 based on correct answers, support this finding. Although there were variations in ability among participants, the distribution of scores tended to center around the mean value.

3.1.2 Data Collection

The data collected in this study included demographic information, paper-based test results, and game logs. We used data from the paper-based test and game logs to analyze performance and estimate participants' abilities, as well as demographic information to complete the participant profile and provide an overview of participant characteristics. Table 2 summarizes the demographic data and participant characteristics.

Table 2. Demographic characteristics and gaming experience of participants.

Characteristics	Response	Frequency	Percentage (%)
Gender	Male	10	45,45
	Female	12	54,55
Age (in years)	11	8	36,36
	12	14	63,64
Educational Level	6th Grade Primary School	22	100
Interest in playing games	Dislike	0	0,00
	Neutral	6	27,27
	Like	16	72,73
Frequency of playing games	Never	5	22,73
	Sometimes	2	9,09
	Frequently	15	68,18
Daily game duration	30 minutes	10	45,45
	1 hour	6	27,27
	>1 hour	6	27,27
Ever learned using games	Yes	21	95,45
	No	1	4,55
Device used for playing games	Smartphone	19	86,36
	Computer/Laptop	2	9,09
	Other	1	4,55

Device ownership	Personal	17	77,27
	Parents	4	18,18
	Sibling/Other	1	4,55

We conducted all data collection and testing activities on a single day during the quiet week following the end of semester exams. The process was conducted simultaneously in one classroom during class time under the supervision of the teacher and the research team to ensure that all participants took the test under consistent environmental conditions, instructions, and implementation procedures. Figure 1 illustrates the data collection and testing procedure employed in this study, encompassing all stages from completing the demographic form to administering the game-based test.

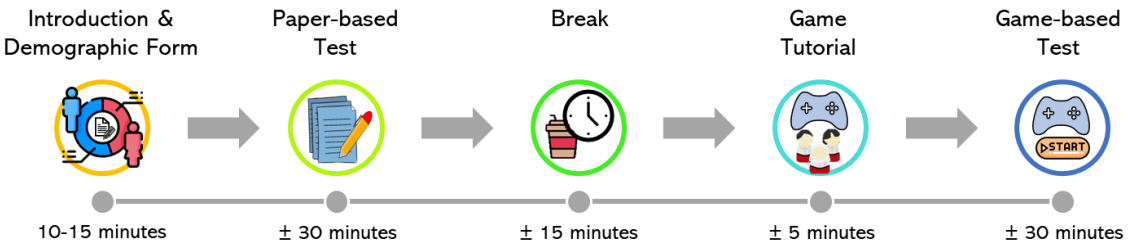

Figure 1. Data collection and testing procedure of the B-Block game study.

First, participants received a general briefing on the research objectives, the stages of the procedure, the testing duration, the principles of data confidentiality, and the consent to participate. After obtaining the information, participants completed a brief demographic form that included the details summarized in Table 2. We confirmed participants' understanding before proceeding to the paper-based test phase. The test lasted approximately 30 minutes and consisted of multiple-choice questions on basic arithmetic operations on integers, aligned with the content of the B-Block game. The mathematics teacher reviewed and validated the paper-based test questions to ensure alignment with the curriculum, and confirmed that the participants had previously learned the four basic arithmetic operations on integers (addition, subtraction, multiplication, and division). After completing the paper-based test, participants took a 15-minute break. Next, they attended a five-minute game tutorial session to understand the game's interface, rules, and mechanics, as well as to gain an overview of its objectives and potential gameplay experiences. Before starting the game-based test session, we reconfirmed participants' understanding of the content presented during the tutorial. Participants then played the B-Block game in test mode for approximately 30 minutes [10]. The system automatically recorded the game logs during the game.

This study used a within-subjects design, where each participant completed both the paper-based test and the game-based test in a fixed order, without applying a counterbalanced design, as in previous studies [10]. Nevertheless, the potential for order effect, such as practice effects or item repetition, was considered minimal due to the design differences between the two types of tests. The paper-based test included questions with a fixed number, order, and difficulty level. In contrast, the game-based test used a procedural approach to generate questions with difficulty levels based on the player's performance in previous levels. Both tests addressed the concept of basic arithmetic operations on integers. However, the problems in the game-based test were adaptive, so their variety and difficulty could differ from those in the paper-based test. Additionally, a brief break was provided between the two tests to minimize the effects of fatigue.

3.2 Description of the B-Block game

The B-Block game was developed to measure the player's performance in solving items related to the four basic arithmetic operations on integers. It was designed to assess players' cognitive skills through puzzle-based challenges [43]. Each level in the game represents an equation item consisting of a target, blocks containing operands and operators, and a number of lives, as shown in Figure 2. The target is the value that players must achieve by combining the blocks using the game mechanics. If the result of the final block does not match the target, players can retry the level by pressing the refresh button. Each time this button is used, the number of lives for that level decreases by one. If the number of lives reaches zero and the player has not achieved the target, the player fails the level, and the game automatically proceeds to the next level. Thus, the number of lives represents the maximum number of attempts a player has to complete a level.

Each level has a varying number of blocks, with a minimum of three and a maximum of five. The operands in the B-Block game are positive or negative integers, with values ranging from 1 to 9. The operators include addition, subtraction, multiplication, and division. The number of multiplication and division operators varies from none to a maximum of four, arranged in various combinations and placed explicitly on the edges of the blocks. Meanwhile, addition and subtraction operators are already integrated in the form of positive or negative operand values. Figure 2 also shows the gameplay interface of one of the levels, consisting of four blocks. Each block contains a positive or negative number and an arithmetic operator. Players operate the numbers by selecting one block first, then selecting a second block to combine them as an operation pair. The result of this operation forms a new block that replaces the two previous blocks. Players repeat this process until all blocks are used up and only one block remains. If the value on the final block matches the displayed target, the level is considered successfully completed.

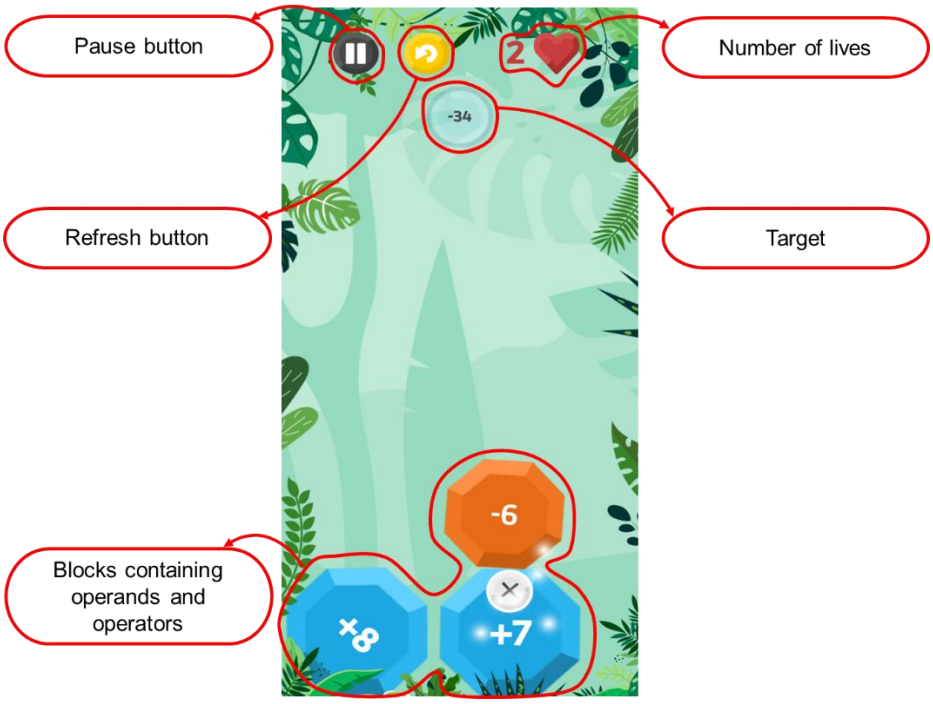

Figure 2. An example of game level and their parts in the B-Block game.

3.3 Proposed method

The proposed method in this study estimates player performance through adaptive puzzle game level generation that is procedurally designed and automatically adjusts the difficulty level based on player achievement. This method includes assessment and game elements

analysis, involving game designers and learning instructors to define the difficulty criteria for items and game levels used in the adaptation process. In the level generation process, the difficulty level is determined adaptively based on player performance prediction, as illustrated in Figure 3.

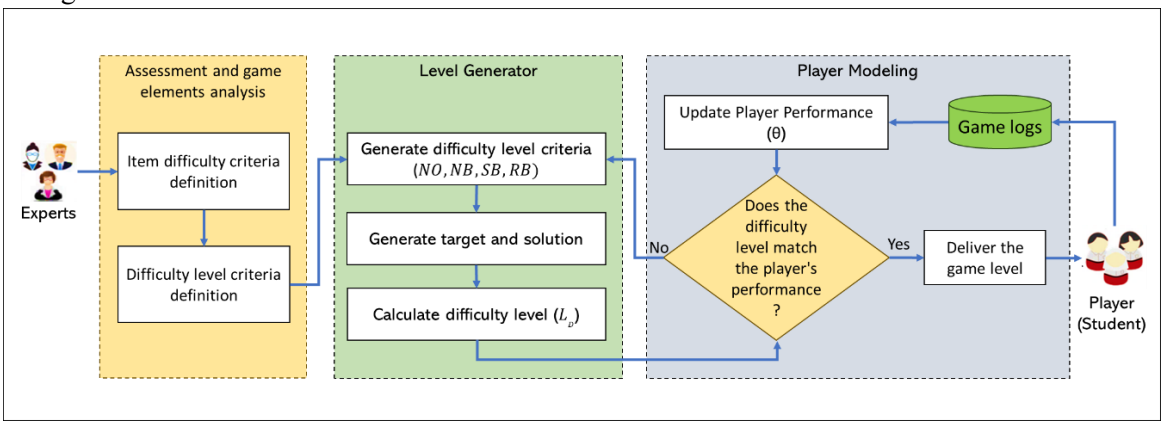

Figure 3. The proposed method.

3.3.1 Assessment and game elements analysis

As described in Section 3.2, the B-Block game consists of several main elements, including targets, blocks containing numbers and operators, and a number of lives mechanism. In the context of GBA, we analyzed these elements in terms of their technical design and their cognitive role in shaping challenges, providing feedback, and supporting adaptive difficulty based on player performance. The analysis ensured that the generated puzzle levels remained relevant to the assessment objectives, enabled valid estimation of player abilities, and provided an engaging game experience. The assessment element analysis identified the item types, difficulty level criteria, possible solutions, and the competencies tested. The game element analysis, on the other hand, covered the target player, game mechanics, level design, and feedback. Together, these processes formed the foundation for adaptive puzzle-level generation, ensuring that game difficulty could be adjusted to the player's performance while applying scaffolding principles in GBA design. In this study, we explicitly distinguished between two dimensions, the cognitive elements of arithmetic items (e.g., number of operands and operator, operator types, operand ranges, and target values) and the technical elements of game interaction (e.g., number and arrangement of blocks, available moves, number of lives, and and interaction rules). We incorporated both dimensions into the adaptive generation process to ensure that level design accurately reflected player performance.

Item difficulty was adjusted dynamically according to the combination of operators and operands, allowing challenges to increase when players succeeded and decrease when they failed, to keep them within an optimal development zone. This adjustment reflects the scaffolding principle, which provides appropriate support to keep players within their developmental range [44]. The approach is grounded in constructivism, emphasizing active knowledge construction [45], and Vygotsky's zone of proximal development (ZPD), which highlights the role of tailored support [46]. In this context, the adaptive system in the game provided challenges and assistance that kept players within this developmental range for assessment purposes.

Based on the description of the B-Block game, the item type used in the game is an equation item. The item difficulty level criteria depend on the item type. For example, the text-based math items have different criteria than equation items in determining the difficulty level [30]. In equation items, the more operator combinations in the item, the higher the complexity. The operator type also affects the item complexity level. Items with addition and subtraction operators tend to have lower complexity than multiplication and division operators [47], [48].

Figure 4 shows the effect of operator combination complexity on item difficulty level. In addition, the more numbers operated, the more the complexity increases. In this study, we defined the item difficulty level criteria as follows:

a. The combination and number of operators,
b. The combination and number of operands (positive and negative numbers),
c. The number of numerical values operated on, and
d. The range of numerical values in the item.

We used the item difficulty level criteria as a foundation to define and determine the difficulty level and its weight (importance) for the game level. We also defined the difficulty level criteria for the game level as follows:

a. The number of blocks generated in each game level (NB),
b. The number of blocks that contain multiplication or division operators (NO),
c. The total number of all operand values (SB), and
d. The range of maximum and minimum operand values (RB).

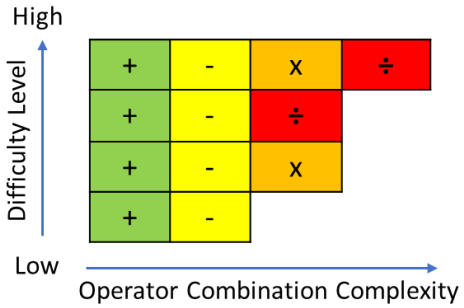

Figure 4. A combination of operators in an item affects the difficulty level.

These difficulty level criteria can vary depending on the game genre and the item types used [25], [27]. In this study, experts determined that NO and NB were the most important criteria, followed by SB and RB. These four criteria served as the basis for determining the difficulty level of automatically generated game levels.

Game element analysis encompasses key aspects, including target players, game mechanics, level design, and feedback. The target players in this study are students who have learned basic arithmetic operations involving integers, as outlined in the applicable curriculum. The game mechanics are intuitive and easy to understand, enabling players to interact by selecting, rotating, or sliding blocks, using the refresh button, and applying the number of lives to complete a level. We created the level design procedurally and adaptively, rather than building it manually as in conventional level design, taking into account the problem form, difficulty parameters, and feedback on player performance. Each level was constructed from a combination of blocks containing operands and operators, with the structure determined by the level's difficulty criteria (NB, NO, SB, RB) and adjusted according to the player's performance in the previous level.

Visual feedback is applied to help players recognize their achievements in each level. When a player successfully completes a level, the last block is displayed in green and accompanied by a confetti effect animation as a visual appreciation for their success. Conversely, if the player fails to reach the target, the system displays the last block in a different color, and the confetti effect animation does not appear (Figure 5). This design reinforces the player's perception of success and failure through direct visual cues. This feedback design not only helped players understand their performance in achieving the objective [34] but also provided immediate feedback based on game activities, enabling the identification of difficulty areas and supporting the creation of an adaptive game environment [12].

Figure 5. Immediate feedback if the player fails to achieve the target (left and center images) and achieves the target (right image) at a game level.

3.3.2 Level generator

The level generator automatically generates puzzle levels of varying difficulty based on predefined difficulty criteria. This process aims to ensure that the generated levels adapt to the player's performance, thus creating a play experience that suits their abilities. Puzzle game-level generation process through the following three main stages:

a. Generate difficulty level criteria

At this stage, the system calculates the value of each variable that represents the difficulty level criteria (NB, NO, SB, and RB). The NB value is the proportion of variations of operand and operator combinations in blocks compared to the number of possible combinations at a level with the maximum number of blocks. Equation (1) calculates NB with the parameter r, which indicates the number of operand and operator combinations in blocks. L_B and M_B are the number of blocks and the maximum number of blocks in the game level. The fewer the number of blocks, the more the NB value tends to be close to 0; the more the number of blocks, the value tends to be close to 1. NB indicates the structure diversity level based on the number of blocks.

$$NB = \frac{C(L_B, r)}{C(M_B, r)} \tag{1}$$

NO represents the ratio between the number of blocks containing multiplication or division operators (O_B) and the number of blocks (L_B) in the game level calculated using equation (2). The NO value is close to 1 if almost all blocks contain multiplication or division operators and zero if there are none.

$$NO = \frac{O_B}{L_B} \tag{2}$$

In the next step, calculate the SB value using equation (3) as the ratio between the total absolute value of all operands in all blocks to the maximum value of operands used in the game (Where O_i is an operand in the ith block in the game level). The maximum value of this operand is obtained by calculating the product of the largest operand in the game (max_{OG}) and the maximum number of blocks (M_B) in the game level. The larger the SB value (closer to 1), the higher the numerical complexity of the resulting level. Thus, the SB

value reflects the number of blocks in the level and the magnitude of the operand value in each block.

$$SB = \frac{\sum_{i=1}^{L_B} |O_i|}{max_{OG}\ M_B} \tag{3}$$

The RB value represents the extent of the range between the absolute largest ($|max_{OL}|$) and smallest ($|min_{OL}|$) value operands in the level, compared to the value of the largest operand in the game (max_{OG}). A high RB value (close to 1) indicates that the range between the largest and smallest operands in the level is relatively small, and a low RB value (close to 0) indicates that the range is getting larger. The RB value is calculated using equation (4).

$$RB = 1 - \left(\frac{|max_{OL}| - |min_{OL}|}{max_{OG}} \right) \tag{4}$$

b. Generate target and solution

Each level of the game should have a target and correct and incorrect solutions. Each solution is designed to achieve that target through a specific sequence of steps, which depends on the number of blocks and operators available at the level. So, each level can have a different solution with different steps in a different order. Algorithm 1 generates a target for each level with a combination of operands and operators based on a specific maximum limit. Then, the various possible solution steps are explored using a tree structure approach to explore the various paths to the target. Figure 6 illustrates this approach: a green leaf represents a correct solution, while a red leaf represents an incorrect solution. The more blocks and operators at a level, the more complex and deep the tree structure becomes as the combination of solutions increases.

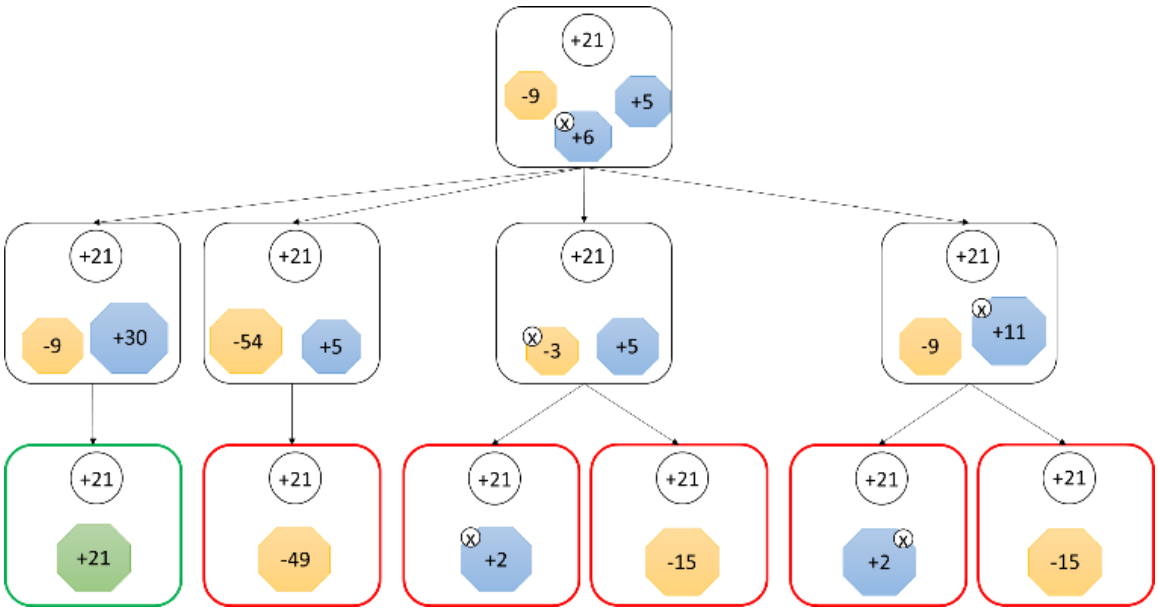

Figure 6. Illustrates solving a puzzle using a tree approach (Green squares are correct targets, and red squares are incorrect targets).

Algorithm 1. Target Generator

Algorithm 1 Target Generator
Input: Set number of block (N), TargetNode (T), maximum combination (M)
Output: Combination of tree from root to target
Initialization: *root node* (r) of tree with N data, empty queue (Q), empty set of TargetNode (E)
1: **While** Q is not empty:
2: **Let** c be the first element on Q
3: **If** | c.data| > M
4: **Let** P be the set of all possible combinations of c.data with at most M elements

```
    End If
    For each combination of p ∈ P:
        Create a new child node v with data of p
        add v as child node from c
        add v to Q
        If p = T Then
            add v to E
        End If
Return E
```

c. Calculate difficulty level

We design the difficulty level (L_D) of a game level by considering several predefined difficulty level criteria and their respective weights (w). The L_D value is calculated using equation (5), which is the sum of the multiplication results between the value of each criterion and its weight, then dividing it by the total weight of all criteria. Each game level has an L_D value from 0 to 1. The higher the L_D value (close to 1), which reflects the more complex and challenging the level is for the player. On the other hand, an L_D value close to 0 indicates a simpler item. This approach allows for a systematic measurement of difficulty, reflecting the complexity of the items within each game level.

$$L_D = \frac{w_{NB}NB + w_{NO}NO + w_{SB}SB + w_{RB}RB}{w_{NB} + w_{NO} + w_{SB} + w_{RB}} \tag{5}$$

3.3.3 Player modeling

The player modeling approach is used to analyze player performance and determine the L_D value of the next level to support adaptive level design. This research applies a model-free approach in DDA, which models player behavior directly based on game data without depending on prior theory or predictive models [14], [49]. Game data, such as the player's performance log, is used to adjust the $L_D$value of the next level adaptively. The system utilizes the player's success and failure patterns to estimate the appropriate difficulty level. In this context, player modeling dynamically identifies player performance (θ) parameters during the game.

The player's ability to complete the game level is represented by θ, which reflects the player's success or failure in achieving the target. This θ value is used as a reference to determine the L_D value in the subsequent level. At the beginning of the game, the θ value is consistently initialized based on the student's grade level [3]. In this context, mechanisms such as the refresh button and the reduction in the number of lives previously described in section 3.2 are used as player performance indicators. Whether success or failure, each outcome contributes to estimating the θ value, subsequently influencing the difficulty adjustment in the next level. The number of lives is integrated into the adaptive model as one of the factors adjusted based on the L_D value.

L_D value is adjusted based on player performance using equation (6). If the player completes the game level successfully, the L_D value increases by adding the variable a to the previous value. Conversely, if the player fails, the variable a value does not change, but the L_D value is reduced by a, or by b if the failure at the beginning of the game. The variable b is a more significant reduction factor when failure occurs in the initial game levels. This more significant decrease in the L_D value at this early stage aims to avoid increasing the difficulty too quickly while providing space for the player to adapt to the game mechanics and demonstrate gradual performance improvement in subsequent levels.

$$\theta = \begin{cases} \text{success}, L_D + a \\ \text{failure} \begin{cases} L_D - b, \text{Initial level} \\ L_D - a \end{cases} \end{cases} \tag{6}$$

The variable a serves as an adjustment factor that controls change in the θ value and reflects the player's capacity to complete the game levels. The a value is calculated using equation (7) by multiplying the scaling constant K with the ratio between the number of remaining retries (R_L) and the number of total retries (R_T). This ratio indicates the proportion of attempts still available to the player and contributes to the θ value adjustment. The constant K serves as a scaling factor to adjust the a value. In this context, R_T represents a player's maximum chances to retry a level using the refresh button, while R_L refers to the number of remaining chances. This approach is similar to the number of lives in games, where the available chances decrease each time the player retries a level. The game automatically proceeds to the next level if all chances are used up. This design ensures that difficulty adjustments remain controlled and that θ values reflect player performance, supporting the generation of levels that adapt to individual abilities.

$$a = K\left(\frac{R_L}{R_T}\right) \tag{7}$$

3.4 Statistical Analysis

In this study, we conducted statistical analysis to evaluate the validity and effectiveness of the proposed GBA, as well as to examine whether the adaptive puzzle level generation method could accurately estimate player performance. A paper-based test was used as a reference standard to evaluate the validity of the game-based test. This test was not used as a pre- or post-test or for a control group, but rather served as the primary benchmark in comparative and correlational analyses. This study treated the paper-based test scores as a representation of students' basic arithmetic operation abilities on integers. We used these scores to ensure that the game-based test measured the same competencies as conventional assessments.

We used descriptive statistics (mean and standard deviation) to summarize the distribution of player performance (θ) scores and level difficulty (L_D) values, providing an overview of the variation in player performance and the variation of levels generated by the level generator. We calculated the Root Mean Square Error (RMSE) to measure the difference between the level difficulty (L_D) values generated by the system and player performance (θ) values. This value represents the accuracy of the adaptive level generator in matching game difficulty to each player's performance. Furthermore, this study performed a Pearson correlation analysis to evaluate the alignment between paper-based test scores and performance estimates from the game. The results of this analysis are presented in Section 4 to demonstrate the performance of the level generator and the system's capability to estimate player performance.

4. Results and Discussion

This study conducted a detailed analysis to evaluate the effectiveness of the proposed adaptive puzzle-level generation approach in estimating player performance and supporting adaptive GBA. In this analysis, we examined the level generator's ability to produce diverse game levels that satisfied predefined difficulty level criteria and to adjust level difficulty based on player performance. We applied statistical methods to assess the accuracy with which the generator estimated player performance.

This study collected game data and paper-based test results to support a comprehensive evaluation. The game data consisted of 1,379 levels, including player ID, player performance (θ), level ID, game mode, difficulty level criteria (NO, NB, SB, RB), difficulty level (L_D), items presented, solutions, number of trials, success/failure status, and playing time. The paper-based test results contained students' answers, the number of correct and incorrect responses, and final scores. This study analyzed all data to evaluate the proposed PCG algorithm's

performance in generating varied game levels that met predefined difficulty criteria and supported estimating player performance.

4.1 Level Generator Performance Analysis

The level generator's ability to produce game levels with varied difficulty, aligned with predefined criteria, requires detailed analysis. Figure 7 (Images 1–10) illustrates ten generated game levels, ranging from the simplest to the most complex. The simplest level (Figure 7, Image 1) consists of three blocks containing integers with addition and subtraction operators. Meanwhile, the most complex levels (Figure 7, Images 8–10) comprise five blocks that contain combinations of integers and operators, including addition, subtraction, multiplication, and division.

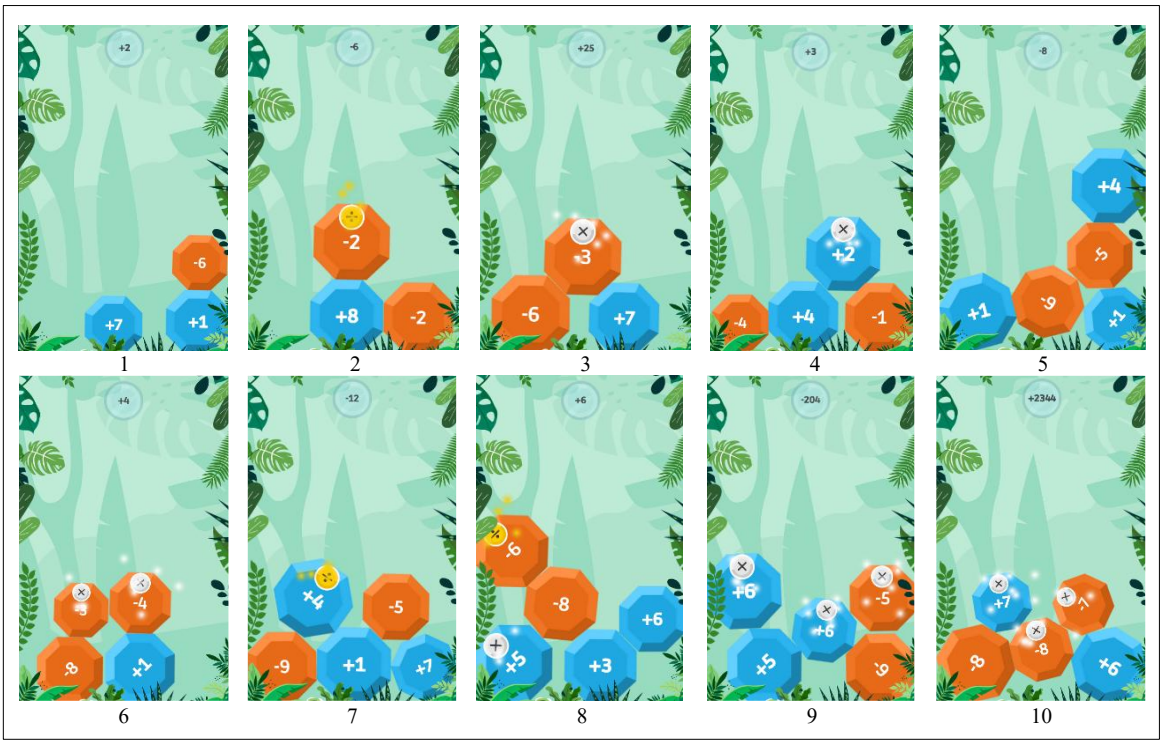

Figure 7. The level generation result: Image 1 shows a game level featuring three blocks that contain integers with addition and subtraction operators. Images 2–3 show game levels with three blocks containing integers with either multiplication or division operators in addition to basic operations. Images 4–10 show game levels with more than three blocks containing integers with combinations of addition, subtraction, multiplication, and division operators.

The generator calculates game level complexity using the L_D value based on the difficulty criteria and the weighted importance of each criterion. Figure 8 shows that every criterion contributes significantly to the L_D value for each game level in Figure 7. For example, the L_D value at level 5 is lower than that at level 6 despite a higher NB score at level 5, because its NO score equals zero. Similarly, the L_D value at level 4 is slightly lower than that at level 5 due to a lower NB score, even though level 4 has a non-zero NO score. These results confirm that the level generator operates as designed, consistently accounting for the weighted contribution of each criterion.

Overall, the level generator successfully produced levels with varied difficulty, determined by the number of blocks, combinations of operands and operators, target values, and operand ranges as specified by the parameters. This finding aligns with previous studies [39]–[41] that automatically generated puzzle levels based on design constraints. However, those studies

[39]–[41] defined difficulty solely in terms of game technical structures, overlooking pedagogical indicators that reflect the conceptual complexity of educational content.

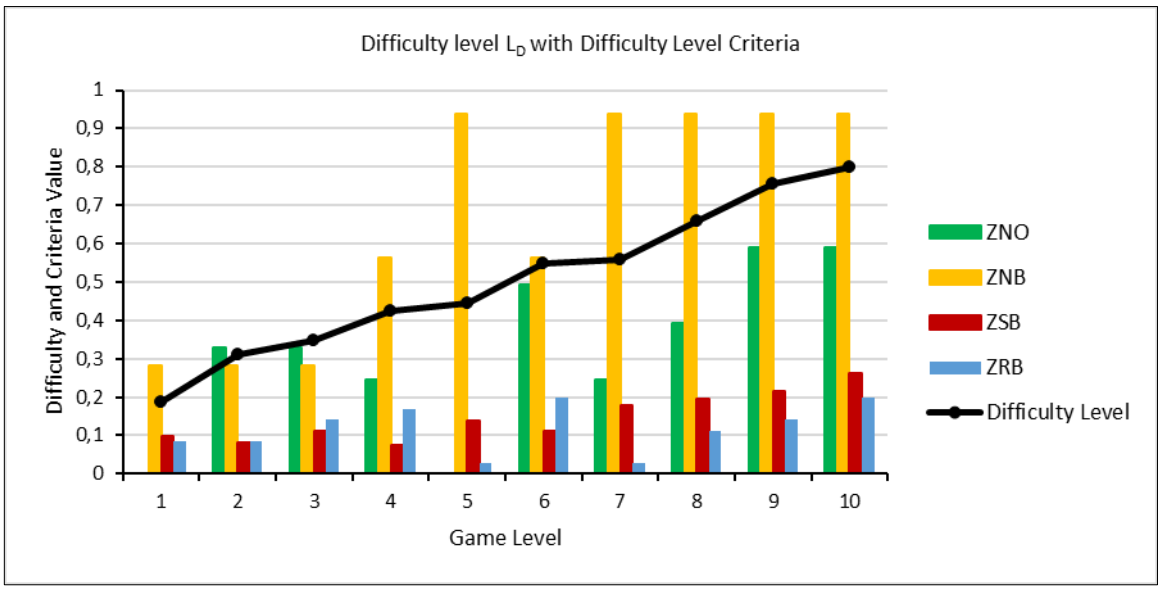

Figure 8. Relationship between L_D and difficulty level criteria (NO, NB, SB, RB) for each game level.

In contrast, the present study defines difficulty through a combination of technical puzzle parameters and mathematical concepts, ensuring that each level varies both structurally and pedagogically in line with the educational content it measures. This approach is comparable to the work of Hooshyar et al. [31], [42], who evaluated generator performance by its ability to produce educational content variations tailored to players' skills based on cognitive difficulty (e.g., complexity of alphabet and phoneme combinations). Extending this concept, our study controls both the complexity of arithmetic problems (content) and puzzle mechanics, producing a wider range of levels that demand cognitive effort and problem-solving strategies. Unlike Hooshyar et al. [31], [42], however, the present approach does not yet incorporate initial ability estimation or players' historical data. As a result, the first level has predetermined difficulty. The system then adaptively determines the difficulty of subsequent levels based on players' performance in the preceding level, within the bounds of the predefined parameter combinations. This design highlights the potential of integrating cognitive and structural complexity to achieve more meaningful GBAs.

4.2 Game Level Difficulty Adjustment Analysis

The adjustment of game level difficulty needs to be analyzed to evaluate how the level generator adjusts L_D values in response to player performance. This analysis also highlights the consistency and effectiveness of these adjustments. We use the difficulty curve to visualize the pattern of changes in L_D values at each game level played by multiple players. Figure 9 presents the curves of three players (P9, P12, and P25), selected to represent variations in performance. The visualization shows that difficulty increased after a successful level and decreased after a failure, confirming that the algorithm dynamically adjusted difficulty.

This pattern, where success leads to an increase in L_D and failure results in a decrease, is further reflected in the average values observed for each player. The average L_D values for completed levels were 0.30 for P9, 0.29 for P12, and 0.32 for P25. These values align with the players' paper-based test results (0.34, 0.28, and 0.48, respectively), indicating that players with higher performance encountered higher L_D levels. These results demonstrate that the

algorithm consistently adapted level difficulty in line with player performance, an approach not implemented in previous studies [39], [40].

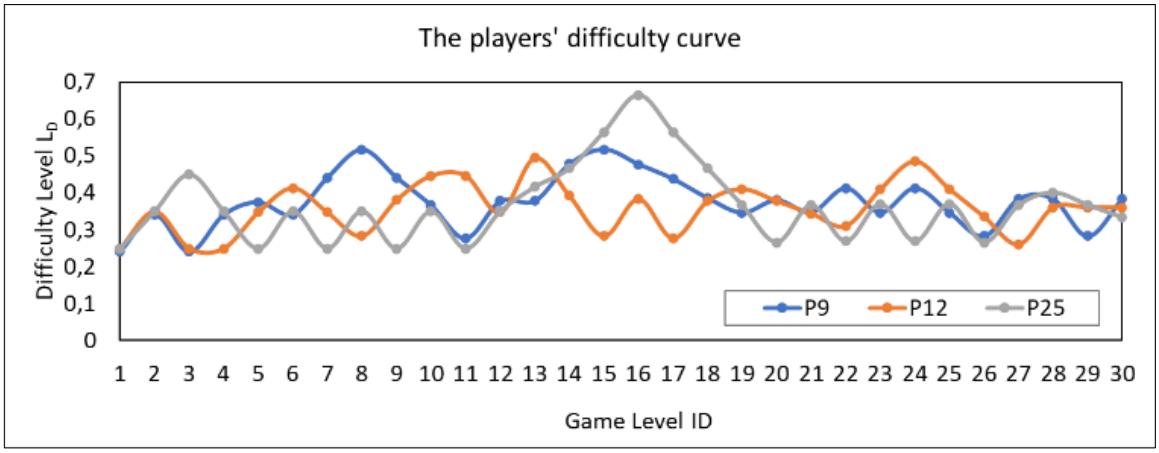

Figure 9. Difficulty curves for three players across 30 initial game levels, illustrating the level generator's ability to adjust subsequent levels based on player performance.

Although this study did not include initial ability estimation for the first level, the level generator was still able to adapt difficulty consistently across diverse player performances. This adaptive capability aligns with a previous study [41], which adjusted difficulty according to player progress. However, that study focused only on technical structures (e.g., components, flow patterns, grid layouts), without considering the conceptual complexity of educational content, as in this study.

In this study, adaptation occurred after players completed each level, differing from Hooshyar et al. [31], [42], who applied adjustments across gameplay phases. Furthermore, their difficulty regulation varied only in educational content (e.g., alphabet–phoneme combinations, number of hints) without altering puzzle mechanics. By contrast, our study integrated variations in both arithmetic content and technical puzzle parameters (number of blocks, operand–operator combinations, target values, number of lives), so that difficulty reflected both structural and pedagogical complexity. Nevertheless, the approaches of Hooshyar et al. [31], [42] remain relevant for future GBA development, particularly for integrating initial ability estimation at the first level.

4.3 Player Performance Estimation Analysis

We analyzed the capability of the level generator to estimate player performance using statistical methods to evaluate the accuracy and consistency of the adaptive puzzle level generation approach in GBA. The analysis included identifying distribution patterns of player performance values (θ) and generated difficulty level (L_D), testing their proximity, and evaluating their relationship with paper-based test results as an external validity. As shown in Figure 10, the frequency distributions of θ and L_D indicate that the algorithm consistently adjusted level difficulty to match player ability, particularly within the ranges of 0.30–0.39 and 0.40–0.49. This pattern demonstrates that the proposed method can accommodate players with diverse abilities by making adaptive and consistent difficulty adjustments based on their performance.

Descriptive statistics in Table 3 show variation in θ values (M = 0.394, SD = 0.102), reflecting differences in player ability, while variation in L_D values (M = 0.404, SD = 0.105) indicates that the generated difficulty levels also varied in line with player ability. The close approximation between L_D and θ confirms that the proposed PCG algorithm effectively adjusted the level difficulty. This finding is supported by the slight absolute difference (M =

0.012, SD = 0.028) and an RMSE of 0.031, indicating that difficulty adjustments were both accurate and consistent.

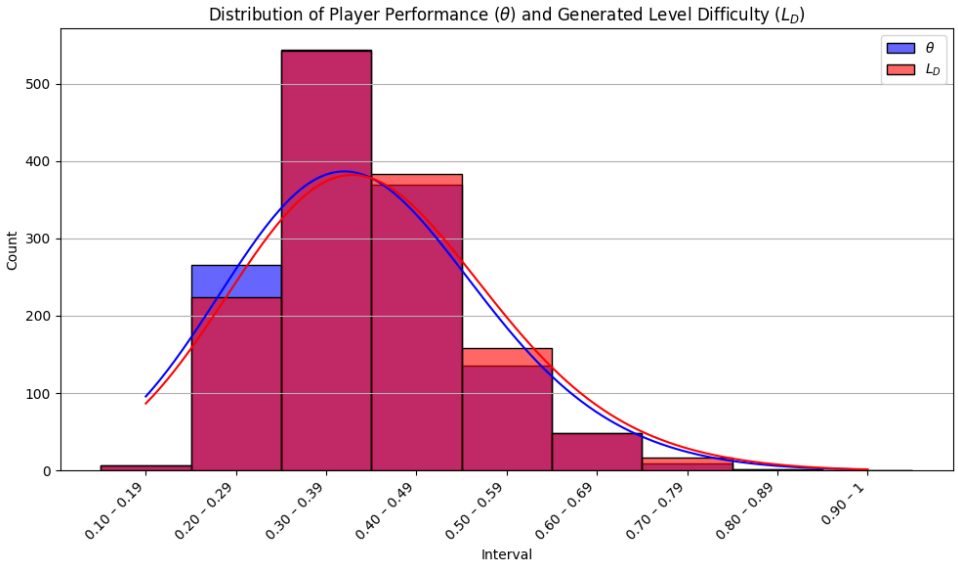

Figure 10. Distribution of player performance (θ) scores and generated level difficulty (L_D) scores across specified intervals.

Table 3. Descriptive statistics based on the θ, L_D, and the absolute difference between θ and L_D.

Variable	Mean	Standard Deviation (SD)
Player Performance (θ)	0.394	0.102
Difficulty Level (L_D)	0.404	0.105
Absolute Difference	0.012	0.028

For external validity, we conducted a correlation analysis between paper-based test scores (number of correct answers) and the average (L_D). values of levels successfully completed by players. The analysis revealed a significant positive correlation ($r = .73$, $p < .001$), indicating that higher paper-based test scores were associated with higher average L_D values. We also examined distribution patterns to assess the effectiveness of the adaptive approach further. As shown in Figure 11, the proposed PCG algorithm successfully estimates player performance by generating levels with difficulty proportional to the player's ability. However, the distributions tended to cluster around specific intervals compared to those from the paper-based test. These findings confirm the effectiveness of the adaptive approach, but also highlight the need to represent ability diversity more effectively. The need for further improvement emphasizes the crucial role of player modeling in accurately linking player ability to level difficulty.

Player modeling plays a key role in estimating performance to determine the difficulty of subsequent levels based on success or failure in earlier ones. A similar approach was employed in a previous study [7] to estimate student competence based on their game interaction histories and predict performance on future tasks. However, that study [7] referred to the method as learner modeling and did not integrate it directly into gameplay. Instead, it applied the method after the game ended to estimate academic competence. Future research should explore more precise rules and formulas to enhance the effectiveness of player modeling in difficulty adaptation.

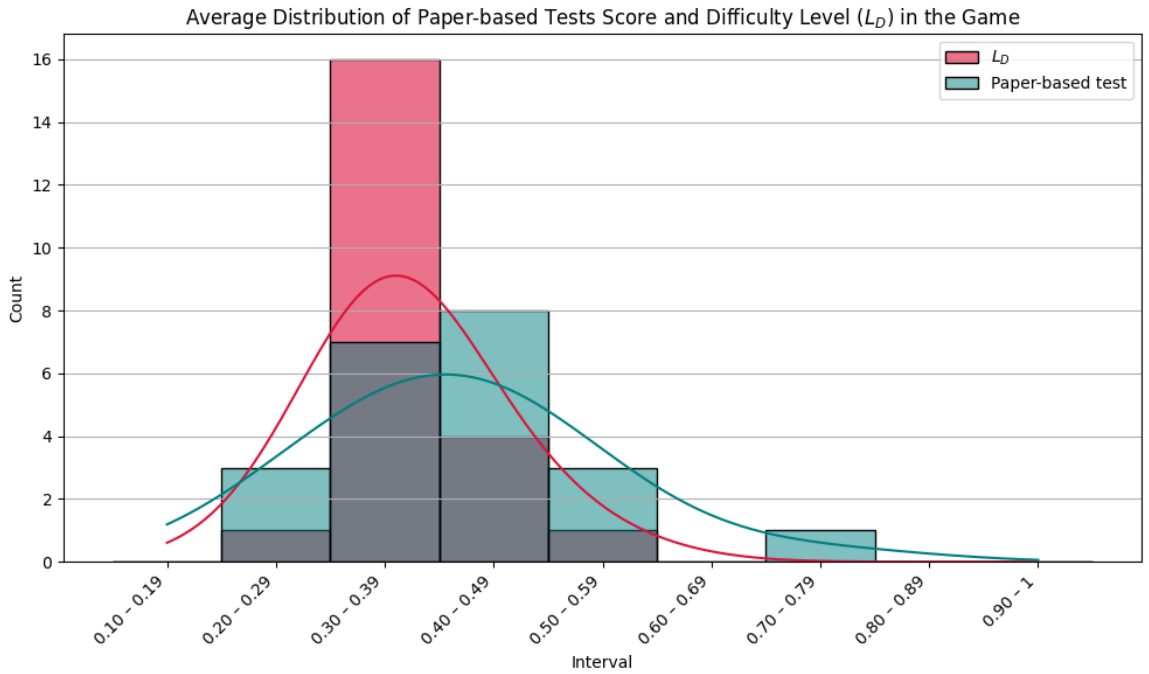

Figure 11. Distribution of average successfully completed difficulty levels and paper-based test scores across specified intervals.

Several PCG studies in educational games [31], [39]–[42] did not conduct external validity to verify whether estimated performance reflected actual ability. Their focus was primarily on developing GBL to measure changes in student ability after gameplay or on evaluating the technical aspects of level generators and gameplay experience using in-game performance metrics. In contrast, this study compared external validity results with mathematics-focused GBA research. Some GBA studies conducted external validity by comparing GBA scores with paper-based test scores [10], [34] or mathematics achievement [35]. Findings from those studies [10], [34], [35] reported significant positive correlations between GBA scores and their comparison instruments, consistent with this study. The main difference is that previous studies developed the game manually, whereas this research utilizes PCG integrated with DDA and player modeling. These results demonstrate that automating level generation still preserves the ability of GBA to estimate player performance. Table 4 summarizes the comparison with related studies, and the findings represent an initial step toward a more comprehensive adaptive GBA model based on real data and performance prediction.

Table 4. Correlations between the paper-based test and GBA based on a previous study.

Research	GBA	PCG	Integrated DDA & player modeling	Comparison instrument	Correlation (r, p-value)
Kiili [10]	Yes	No	No	Paper-based test	0.73 (<.001)
Kiili [34]	Yes	No	No	Paper-based test	0.79 (<.001)
Ninaus [35]	Yes	No	No	General math achievement	• Fraction comparison performance: 0.57 ($p < .001$) • Fraction estimation accuracy: 0.71 ($p < .001$)
Proposed method	**Yes**	**Yes**	**Yes**	**Paper-based test**	**0.73 (<.001)**

5. Conclusions

This study aimed to estimate player performance through adaptive puzzle-based level generation in GBA. The proposed method transformed assessment standards into level difficulty variables, procedurally generated levels that reflected player performance, and reliably estimated ability by aligning difficulty with both cognitive (arithmetic items) and technical (puzzle parameters) elements. Statistical analyses confirmed the effectiveness of this method, as evidenced by strong positive correlations between paper-based test scores and the average difficulty of successfully completed levels. A low RMSE and small absolute differences between difficulty level (L_D) and player performance (θ) further demonstrated the accuracy and consistency of difficulty adaptation.

The findings highlight broader implications for educational game design and player experience. For educational game developers, the study offers practical recommendations for calibrating technical puzzle parameters and cognitive difficulty, ensuring that adaptive levels remain pedagogically meaningful. From the player's perspective, adaptive difficulty ensures that challenges remain engaging and aligned with their abilities, thereby supporting a more balanced and meaningful gameplay experience. In the educational context, adaptive GBA can provide automatic estimates of student ability that complement conventional assessments. In addition, adaptive GBA expands opportunities for self-directed learning, as students can engage with practice exercises without waiting for teachers to provide them. While this study focused on arithmetic, the same principles can be extended to other domains, such as language or science, by determining difficulty level criteria and aligning with the cognitive characteristics of each subject.

However, several limitations remain. The distribution of generated difficulty levels requires further optimization to capture a broader range of player abilities, and the small sample size limits the generalizability of the results. Moreover, this study did not incorporate teachers' perspectives on integrating GBA into classroom practice. Future research should explore the incorporation of initial ability estimation or historical player data, investigate more sophisticated player modeling algorithms, and involve larger and more diverse participant groups to strengthen external validity. With these extensions, the proposed PCG-based adaptive GBA method can become more effective, personalized, and relevant for future innovative GBAs.

Conflicts of interest

The authors declare no conflicts of interest and affirm that there are no commercial or other interests that could have influenced the outcomes or integrity of this study.

Ethical approval and consent to participate

This study was approved by the Research Ethics Committee of the Faculty of Psychology, Muhammadiyah University of Malang (Approval No. E.6.m/251/KE-FPsi-UMM/I/2025), in compliance with applicable government regulations. The study adhered to the principles of beneficence, fairness, and respect for persons, and met seven ethical standards as well as 25 CIOMS-WHO guidelines covering aspects such as social and clinical value, scientific design, informed consent, and participant privacy.

Acknowledgments

This work was supported by the Lembaga Pengelola Dana Pendidikan (LPDP), Ministry of Finance, Republic of Indonesia. We also thank SDN 8 Mataram (West Nusa Tenggara, Indonesia) for their invaluable support and facilitation throughout the research process.

References

[1] Y. J. Kim and D. Ifenthaler, "Game-Based Assessment: The Past Ten Years and Moving Forward," *Game-Based Assess. Revisit.*, pp. 3–11, 2019, doi: 10.1007/978-3-030-15569-8_1.

[2] P. Black and D. Wiliam, "Assessment and classroom learning," *Int. J. Phytoremediation*, vol. 21, no. 1, pp. 7–74, 1998, doi: 10.1080/0969595980050102.

[3] U. L. Yuhana, E. M. Yuniarno, W. Rahayu, and E. Pardede, "A Context-based Question Selection Model to Support the Adaptive Assessment of Learning: A study of online learning assessment in elementary schools in Indonesia," *Educ. Inf. Technol.*, vol. 29, no. 8, pp. 9517–9540, 2024, doi: 10.1007/s10639-023-12184-8.

[4] C. Udeozor, P. Chan, F. Russo Abegão, and J. Glassey, "Game-based assessment framework for virtual reality, augmented reality and digital game-based learning," *Int. J. Educ. Technol. High. Educ.*, vol. 20, no. 1, 2023, doi: 10.1186/s41239-023-00405-6.

[5] A. C. M. Yang, B. Flanagan, and H. Ogata, "Adaptive formative assessment system based on computerized adaptive testing and the learning memory cycle for personalized learning," *Comput. Educ. Artif. Intell.*, vol. 3, no. October, p. 100104, 2022, doi: 10.1016/j.caeai.2022.100104.

[6] Sein Minn, "AI-assisted knowledge assessment techniques for adaptive learning environments," *Comput. Educ. Artif. Intell.*, vol. 3, no. July 2021, p. 100050, 2022, doi: 10.1016/j.caeai.2022.100050.

[7] J. e A. Ruiperez-Valiente, Y. J. Kim, R. S. Baker, P. A. Mart, and G. C. Lin, "The Affordances of Multivariate Elo-Based Learner Modeling in Game-Based Assessment," *IEEE Trans. Learn. Technol.*, vol. 16, no. 2, pp. 152–165, 2023, doi: 10.1109/TLT.2022.3203912.

[8] S. Elyasi, A. Varastehnezhad, and F. Taghiyareh, "From Play to Prediction: Assessing Depression and Anxiety in Players Behavior with Machine Learning Models," *Int. J. Serious Games*, vol. 12, no. 1, pp. 83–102, 2025, doi: 10.17083/ijsg.v12i1.897.

[9] D. W. Putwain and W. Symes, "The Four Ws of Test Anxiety: What is it, why is it important, where does it come from, and what can be done about it?," *Psychologica*, vol. 63, no. 2, pp. 31–52, 2020, doi: 10.14195/1647-8606_63-2_2.

[10] K. Kiili and H. Ketamo, "Evaluating Cognitive and Affective Outcomes of a Digital Game-Based Math Test," *IEEE Trans. Learn. Technol.*, vol. 11, no. 2, pp. 255–263, 2018, doi: 10.1109/TLT.2017.2687458.

[11] X. Ren, "Stealth Assessment Embedded in Game- Based Learning to Measure Soft Skills: A Critical Review," *Game-Based Assess. Revisit.*, pp. 67–83, 2019, doi: 10.1007/978-3-030-15569-8_8.

[12] M. J. Gomez, J. A. Ruiperez-Valiente, and F. J. G. C. Clemente, "A Systematic Literature Review of Game-Based Assessment Studies: Trends and Challenges," *IEEE Trans. Learn. Technol.*, vol. 16, no. 4, pp. 500–515, 2023, doi: 10.1109/TLT.2022.3226661.

[13] J. C. Perry, S. Balasubramanian, C. Rodriguez-de-pablo, and T. Keller, "Improving the match between ability and challenge : toward a framework for automatic level adaptation in game-based assessment and training," in *2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR)*, 2013, pp. 1–6, doi: 10.1109/ICORR.2013.6650420.

[14] P. Shi and K. Chen, "Learning constructive primitives for real-time dynamic difficulty adjustment in super Mario bros," *IEEE Trans. Games*, vol. 10, no. 2, pp. 155–169, 2018, doi: 10.1109/TCIAIG.2017.2740210.

[15] F. Ke, B. Parajuli, and D. Smith, "Assessing Game-Based Mathematics Learning in Action," *Game-Based Assess. Revisit.*, pp. 213–227, 2019, doi: 10.1007/978-3-030-15569-8_8.

[16] F. Schubert, M. Awiszus, and B. Rosenhahn, "TOAD-GAN: A Flexible Framework for Few-Shot Level Generation in Token-Based Games," *IEEE Trans. Games*, vol. 14, no. 2, pp. 284–293, 2022, doi: 10.1109/TG.2021.3069833.

[17] T. Gao, J. Zhang, and Q. Mi, "Procedural Generation of Game Levels and Maps: A Review," *4th Int. Conf. Artif. Intell. Inf. Commun. ICAIIC 2022 - Proc.*, pp. 50–55, 2022, doi: 10.1109/ICAIIC54071.2022.9722624.

[18] Y. Zakaria, M. Fayek, and M. Hadhoud, "Procedural Level Generation for Sokoban via Deep Learning: An Experimental Study," *IEEE Trans. Games*, vol. 1502, no. c, pp. 1–13, 2022, doi:

10.1109/TG.2022.3175795.
[19] G. N. Yannakakis and J. Togelius, "Experience-driven procedural content generation," *IEEE Trans. Affect. Comput.*, vol. 2, no. 3, pp. 147–161, 2011, doi: 10.1109/T-AFFC.2011.6.
[20] S. P. Walton, A. A. M. Rahat, and J. Stovold, "Evaluating Mixed-Initiative Procedural Level Design Tools Using a Triple-Blind Mixed-Method User Study," *IEEE Trans. Games*, vol. 14, no. 3, pp. 413–422, 2022, doi: 10.1109/TG.2021.3086215.
[21] Z. Zhou, Z. Lu, M. Guzdial, and F. Goes, "Creativity Evaluation Method for Procedural Content Generated Game Items via Machine Learning," *Proc. - 2022 9th Int. Conf. Dependable Syst. Their Appl. DSA 2022*, pp. 249–253, 2022, doi: 10.1109/DSA56465.2022.00042.
[22] A. Liapis, G. N. Yannakakis, and J. Togelius, "Limitations of choice-based interactive evolution for game level design," *AAAI Work. - Tech. Rep.*, vol. WS-12-17, pp. 33–36, 2012, doi: 10.1609/aiide.v8i5.12571.
[23] M. Pichlmair and M. Johansen, "Designing Game Feel: A Survey," *IEEE Trans. Games*, vol. 14, no. 2, pp. 138–152, 2022, doi: 10.1109/TG.2021.3072241.
[24] J. Roberts, K. Chen, and S. Member, "Learning-Based Procedural Content Generation," *IEEE Trans. Comput. Intell. AI GAMES*, vol. 7, no. 1, pp. 88–101, 2015, doi: 10.1109/TCIAIG.2014.2335273.
[25] Ł. Spierewka, R. Szrajber, and D. Szajerman, "Procedural Level Generation with Difficulty Level Estimation for Puzzle Games," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12746 LNCS, pp. 106–119, 2021, doi: 10.1007/978-3-030-77977-1_9.
[26] S. G. Nam, C. H. Hsueh, and K. Ikeda, "Generation of Game Stages With Quality and Diversity by Reinforcement Learning in Turn-Based RPG," *IEEE Trans. Games*, vol. 14, no. 3, pp. 488–501, 2022, doi: 10.1109/TG.2021.3113313.
[27] M. Van Kreveld, M. Loffler, and P. Mutser, "Automated puzzle difficulty estimation," in *2015 IEEE Conference on Computational Intelligence and Games, CIG 2015 - Proceedings*, 2015, pp. 415–422, doi: 10.1109/CIG.2015.7317913.
[28] A. Summerville *et al.*, "Procedural content generation via machine learning (PCGML)," *IEEE Trans. Games*, vol. 10, no. 3, pp. 257–270, 2018, doi: 10.1109/TG.2018.2846639.
[29] B. de Kegel and M. Haahr, "Procedural puzzle generation: A survey," *IEEE Trans. Games*, vol. 12, no. 1, pp. 21–40, 2020, doi: 10.1109/TG.2019.2917792.
[30] Y. Xu, R. Smeets, and R. Bidarra, "Procedural generation of problems for elementary math education," *Int. J. Serious Games*, vol. 8, no. 2, pp. 49–66, 2021, doi: 10.17083/ijsg.v8i2.396.
[31] D. Hooshyar, M. Yousefi, and H. Lim, "A Procedural Content Generation-Based Framework for Educational Games: Toward a Tailored Data-Driven Game for Developing Early English Reading Skills," *J. Educ. Comput. Res.*, vol. 56, no. 2, pp. 293–310, 2018, doi: 10.1177/0735633117706909.
[32] F. Leutner, S. C. Codreanu, J. Liff, and N. Mondragon, "The potential of game- and video-based assessments for social attributes: examples from practice," *J. Manag. Psychol.*, vol. 36, no. 7, pp. 533–547, 2021, doi: 10.1108/JMP-01-2020-0023.
[33] V. Shute *et al.*, "Maximizing learning without sacrificing the fun: Stealth assessment, adaptivity and learning supports in educational games," *J. Comput. Assist. Learn.*, vol. 37, no. 1, pp. 127–141, 2021, doi: 10.1111/jcal.12473.
[34] K. J. M. Kiili, K. Devlin, A. Perttula, P. Tuomi, and A. Lindstedt, "Using video games to combine learning and assessment in mathematics education," *Int. J. Serious Games*, vol. 2, no. 4, 2015, doi: 10.17083/ijsg.v2i4.98.
[35] M. Ninaus, K. Kiili, J. McMullen, and K. Moeller, "Assessing fraction knowledge by a digital game," *Comput. Human Behav.*, vol. 70, pp. 197–206, 2017, doi: 10.1016/j.chb.2017.01.004.
[36] Y. L. Chien *et al.*, "Game-Based Social Interaction Platform for Cognitive Assessment of Autism Using Eye Tracking," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 749–758, 2023, doi: 10.1109/TNSRE.2022.3232369.
[37] M. Cutumisu, D. B. Chin, and D. L. Schwartz, "A digital game-based assessment of middle-school and college students' choices to seek critical feedback and to revise," *Br. J. Educ. Technol.*, vol. 50, no. 6, pp. 2977–3003, Nov. 2019, doi: 10.1111/bjet.12796.
[38] V. Vallejo *et al.*, "Evaluation of a novel Serious Game based assessment tool for patients with Alzheimer's disease," *PLoS One*, vol. 12, no. 5, p. e0175999, May 2017, doi: 10.1371/journal.pone.0175999.
[39] L. Rodrigues and J. Brancher, "Procedurally generating a digital math game's levels: Does it impact players' in-game behavior?," *Entertain. Comput.*, vol. 32, no. October, p. 100325, 2019, doi: 10.1016/j.entcom.2019.100325.
[40] L. Rodrigues, R. P. Bonidia, and J. D. Brancher, "A Math Educacional Computer Game Using Procedural Content Generation," in *Anais do XXVIII Simpósio Brasileiro de Informática na Educação (SBIE 2017)*, 2017, p. 756, doi: 10.5753/cbie.sbie.2017.756.

[41] A. M. Smith, E. Andersen, M. Mateas, and Z. Popović, "A case study of expressively constrainable level design automation tools for a puzzle game," *Found. Digit. Games 2012, FDG 2012 - Conf. Progr.*, no. c, pp. 156–163, 2012, doi: 10.1145/2282338.2282370.
[42] D. Hooshyar, M. Yousefi, M. Wang, and H. Lim, "A data-driven procedural-content-generation approach for educational games," *J. Comput. Assist. Learn.*, vol. 34, no. 6, pp. 731–739, Dec. 2018, doi: 10.1111/jcal.12280.
[43] M. N. Azzmi.H., U. L. Yuhana, N. Sulistyani, and L. Husniah, "Analyzing the Quality of Game-based Assessment Design in Basic Arithmetic Operations," *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, vol. 4, no. 1, 2023, doi: 10.22219/kinetik.v8i1.1599.
[44] J. M. Baalsrud Hauge *et al.*, "Learning Analytics Architecture to Scaffold Learning Experience through Technology-based Methods," *Int. J. Serious Games*, vol. 2, no. 1, 2015, doi: 10.17083/ijsg.v2i1.38.
[45] M. Larochelle, N. Bednarz, and J. Garrison, Eds., "Constructivism and Education," *Constr. Educ.*, Aug. 1998, doi: 10.1017/CBO9780511752865.
[46] W. M. Roth and L. Radford, "Re/thinking the zone of proximal development (symmetrically)," *Mind, Cult. Act.*, vol. 17, no. 4, pp. 299–307, 2010, doi: 10.1080/10749031003775038.
[47] National Governors Association Center for Best Practices and Council of Chief State School Officers, "Common core state standards for mathematics," *Washington, DC: NGA Center & CCSSO*, 2010. [Online]. Available: http://www.corestandards.org/. [Accessed: 19-Mar-2025].
[48] U. L. Yuhana *et al.*, "A rule-based expert system for automatic question classification in mathematics adaptive assessment on indonesian elementary school environment," *Int. J. Innov. Comput. Inf. Control*, vol. 15, no. 1, pp. 143–161, 2019, doi: 10.24507/ijicic.15.01.143.
[49] G. N. Yannakakis and J. Togelius, "Artificial intelligence and games," *Artif. Intell. Games*, pp. 1–337, 2018, doi: 10.1007/978-3-319-63519-4.