Annales Mathematicae et Informaticae
Accepted manuscript
DOI: 10.33039/ami.2024.12.001
URL: https://ami.uni-eszterhazy.hu

Simulation-driven optimisation in didactic game design*

Enikő Palencsár[a], **Szilvia Szilágyi**[b]

[a]Faculty of Mechanical Engineering and Informatics,
University of Miskolc
palencsar.eniko@student.uni-miskolc.hu

[b]Department of Analysis, Institute of Mathematics,
University of Miskolc
szilvia.szilagyi@uni-miskolc.hu

Abstract. In educational settings, the utilisation of didactic games is a growing trend. Modifying existing games often proves inadequate for addressing certain complex course materials, necessitating the development of original didactic games with novel sets of rules. The design of such games is a challenging endeavour fraught with potential pitfalls. The fine-tuning of new games typically involves extensive trial and error, a process that is both time-consuming and labour-intensive. However, Monte Carlo simulations offer a time-efficient alternative for determining the optimal values of numerical game parameters. This paper illustrates this approach through the example of the YETI cooperative board game, which has the direct comparison test of infinite series at its didactic focus. First, the three levels of game design are briefly introduced as defined by the MDA framework. Drawing from this model, the tuning process of the game is presented, involving a systematic, cluster-based approach to marking the infinite sums featured in the YETI card decks, several strategies to mitigate the impact of the Alpha Player Problem in the game, and the configuration of key game parameters via simulations.

Keywords: game design, MDA model, Monte Carlo simulation, didactic game, infinite series

AMS Subject Classification: 97D40, 65C05, 40A05

*This research was supported by the ÚNKP-23-1 New National Excellence Program of the Ministry for Culture and Innovation from the Source of the National Research, Development and Innovation Fund.

Submitted: June 22, 2024
Accepted: December 12, 2024
Published online: December 13, 2024

1. Introduction

Didactic game design plays an important role in developing innovative approaches for teaching mathematics. Game-based learning is a widespread educational method that is gaining more and more prominence due to the preference of younger generations for hands-on learning experiences [11, 20, 30]. Games can engage students, help them maintain their attention, boost their motivation, and improve their skills while simultaneously promoting teamwork and communication, which are essential skills in adulthood. Researchers often regard digital games as the primary medium for educational purposes. Nonetheless, traditional tabletop games with tangible components offer considerable advantages as well, such as fostering collaboration, or allowing for smooth adjustments and overall control of the gaming experience [3, 40]. Non-digital games also have the potential to stimulate learners' creativity, memory, empathy, problem-solving skills, and self-confidence [31].

The design of educational games is guided by several key principles and shaped by the creators' decisions throughout the process. Didactic board game design can be challenging yet rewarding. At times, it entails creating entirely new games, while on other occasions, it involves adapting existing games for educational purposes. In the context of games, modding is defined as "using existent commercial games and adapting their mechanisms, narratives, rules, and components to achieve specific goals" [32]. For instance, LimStorm is a card game that follows the rules of SOLO, designed as an educational tool for practising the topic of limits [34]. The original game was modified to integrate the intended didactic content by replacing the numbers on traditional SOLO cards with limit expressions. Another example of an adaptation is GemHunters, which originated from the well-known board game Saboteur and aims to help players master the use of certain trigonometric functions and identities [7]. In the modded version of Saboteur, path cards include additional trigonometric formulas, the values of which must form monotonically increasing or decreasing sequences on each assembled path. Generally speaking, modded games can often be constructed more quickly than original games. Their design process typically involves fewer logical errors, though didactic flaws can still occur. However, adaptations are constrained by the structure and rules of their source material, offering limited flexibility compared to the invention of original games.

On the other hand, when no suitable game exists for adaptation, educators can design original games tailored to the needs and preferences of their students. This intricate process is prone to errors and miscalculations, necessitating a methodology that allows for continuous and thorough testing from the beginning. For more complex games, separate tests for different components may be required. However, the additional effort is often worthwhile, as original games offer significantly greater flexibility and wider customisation options compared to adaptations. One such didactic game is YETI, designed to familiarise university students with the concept of infinite series and the application of their direct comparison test.

This paper presents the application of Monte Carlo simulations in game design as a time-efficient method for fine-tuning numerical game parameters, using the example of YETI as a case study. In the next section, a literature review is provided of relevant research on game design and the application of the Monte Carlo method. Subsequently, the board game YETI is briefly introduced, followed by the presentation of its tuning process, which encompasses the selection of the infinite sums featured in the YETI card decks, and the calibration of the most important game parameter values, as well as the exploration of different game variants for addressing the Alpha Player Problem.

2. Literature review

Tabletop game design presents a myriad of challenges that must be addressed. Unlike other forms of entertainment such as books, music, or movies, the consumption of games is unpredictable [14]. When a game is being developed, the specific events and outcomes that will unfold during gameplay are unknown to the designer, which complicates the design process. An important aspect of game development is user-centred design, where the game must provide an interactive experience in which the player exercises agency and autonomy [41]. For a game design attempt to be considered successful, it must harmonise the three key elements of the gaming experience: the designer's intentions, the game artifact providing said experience, and the impressions of the player who engages with the game [41].

Hunicke, Leblanc, Zubek [14] identify three levels of abstraction in their MDA model for game design. Mechanics refer to the fundamental activities and mechanisms available to the player, while dynamics describe the real-time behaviour of these mechanics during gameplay. The third level, aesthetics, encompasses the emotional responses elicited in the player during interactions with the game system. From the designer's viewpoint, mechanics define the boundaries of dynamic behaviour, which in turn shapes the aesthetic experience. On the other hand, the players first connect with the aesthetics of the game, which are expressed through observable dynamics and mechanics. Mechanics consist of manipulable game components, permissible actions, and the rules governing these actions. For instance, the mechanics of card games include shuffling and betting, from which dynamics like bluffing can originate [14]. Adjusting game mechanics has a direct impact on game dynamics. Thus, developing models that predict and describe dynamics can help avoid common design errors.

In a revision of the MDA model, Zubek [41] makes a distinction between mechanics, gameplay – the process of players interacting with game mechanics – and player experience, noting that the term aesthetics is often misunderstood to refer solely to visual appeal rather than the entire experience. Zubek [41] also points out that positioning the designer and the player at opposite ends of the MDA chain may be misleading, as the iterative design process involves both top-down and bottom-up approaches, simultaneously addressing game design from both perspectives. Top-down design begins with the desired experience and deconstructs it into various components, identifying the type of gameplay needed to generate this experience and the mechanics required to produce this gameplay. Each game aims to achieve multiple player experience goals to varying extents, such as providing enjoyable sensory experiences (sensation), transporting players into a different world where they can engage in activities that are impossible or unlikely in real life (fantasy), inducing challenge through time pressure and competition, fostering fellowship through teamwork and information sharing, and creating dramatic tension through a compelling narrative [14].

On the other hand, bottom-up design consists of developing game mechanics, testing them with real players, and continuously evaluating the resulting player experience. Upton [37] provides six important heuristics for assessing and improving game mechanics. The initial two heuristics, choice and variety, highlight that players must perceive a diverse array of possible actions. A limited range of actions can lead to boredom due to the repetitive nature of the game, while an excess of choices can result in confusion and frustration. The third heuristic, consequence, emphasises the importance of player actions having tangible outcomes, given that players may feel a loss of agency within the game in the absence of direct consequences. The fourth heuristic is predictability, followed by uncertainty as the fifth, indicating that players must be able to foresee the outcomes of their actions to a certain extent, but these consequences must not be completely predetermined. The final heuristic, satisfaction, suggests that in a game of high quality, desirable outcomes must always be within reach. An unwinnable game can lead to frustration and disinterest, whereas a game that always delivers desirable outcomes with minimal effort can lack challenge and excitement. Hence, balancing difficulty and attainability is essential for sustaining players' interest and engagement [37].

Balancing, which aims to boost game quality through numerous iterations of fine-tuning and playtesting [17], is among the most challenging phases of game design. During this process, designers rely on adjustable game parameters to reach the intended player experience. Each unique combination of game parameter values can be regarded as a distinct vector in a multidimensional space of game variants, called the game space [16, 36]. It is important to note that this iterative tuning approach can be both costly and time-consuming. In game-based learning, the need to harmonise educational content with gameplay mechanics introduces an additional layer of complexity to the balancing process [4].

The ability to explore the game space without relying entirely on human testers can significantly accelerate the tuning process, as automated methods that facilitate the identification of the most promising configurations reduce the need for extensive playtesting. Jaffe et al. [17] raise the possibility of automating some aspects of game evaluation using AI-driven simulation tools. Similarly, Nelson [21] suggests that metrics derived directly from the game itself, rather than empirical playtests, can offer valuable insights. Hypothetical player-testing, as described by Nelson [21], is an evaluation strategy that uses simplified player models to analyse how the game behaves in idealised or extreme scenarios. Prior studies imply that this approach can effectively reduce the need for player-intensive testing [2, 18, 19]. Chaslot et al. [6] argue that game AI often requires extensive domain knowledge and long development cycles. However, Monte Carlo-based solutions can counter these challenges by using simulated playouts instead of domain-specific heuristics.

The Monte Carlo method, named after the Monte Carlo Casino in Monaco, relies on probabilistic models to estimate the average characteristics of real-world processes [27]. This approach is particularly valuable if direct experimentation is too time-consuming, impractical, or costly [10]. John von Neumann and Stanislaw Ulam pioneered Monte Carlo simulations during the Manhattan Project to model the random diffusion of neutrons in nuclear materials [28]. The application of the method consists of repeated random sampling and statistical analysis. It approximates the value of an unknown quantity using the principles of inferential statistics, which assert that a random sample – a proper subset of a population – tends to reflect the properties of the population.

Monte Carlo simulation typically involves several steps. First, a deterministic model, closely resembling the real scenario, is generated using the most likely values of input parameters, and mathematical relationships to transform the input values into the desired outputs. Once the deterministic model is satisfactory, risk components are added by identifying the underlying distributions of the input variables, based on historical data. Subsequently, random samples are drawn from these distributions representing various sets of input values, which are used in the deterministic model to generate sets of output values. This process is repeated to collect a range of possible outputs. Finally, statistical analysis is performed on these output values to provide a basis for decision-making with statistical confidence [26]. Confidence in the estimate depends on both the size and the variance of the sample: higher variance necessitates larger samples to attain the same level of confidence. By leveraging simulations, researchers can investigate complex systems, repeat experiments, and make modifications as needed. However, there are limitations to using simulations as a modelling methodology. Instead of providing exact measurements, simulations yield statistical estimates, leading to uncertain results prone to experimental errors [28]. Additionally, simulations can be computationally intensive, and the accuracy of the results depends heavily on the quality of the model and the inputs used. Furthermore, like any software, simulation programs can have bugs [10].

In the context of game design, according to Nummenmaa, Kuittinen, Holopainen [24], simulations aim to abstract models to a degree where designers can focus on the core dynamics of a system without being overwhelmed by small details. This approach is particularly advantageous, as simulations can uncover potential issues in the long-term dynamics of a game. Beyond diagnostics, simulations can assist in the tuning of game parameter values during the balancing process. They also serve as a powerful tool for analysing existing games, contributing to the invention of optimal strategies, the discovery of new game variants, and the development of AI-driven players. Additionally, Schell [29] suggests that Monte Carlo simulations can be valuable in assessing the role of chance within a non-deterministic game.

Numerous examples highlight the versatility of Monte Carlo methods in game design. Browne, Maire [5] employed simulations to assess the quality of turn-based, deterministic games and to discover new, high-quality game variants. Similarly, Isaksen et al. [15] applied Monte Carlo simulations to explore the game space in *Flappy Bird*, uncovering a wide range of playable configurations. A specific implementation of Monte Carlo methods, Monte Carlo Tree Search (MCTS), is a best-first search method designed to pinpoint the most promising moves in a given game scenario [39]. In addition to its applications in persona-based player modelling [12] and skill-based automated playtesting [13], MCTS has proven to be a valuable tool in research related to deterministic games, such as *Go* [9] and *Hex* [1], as well as games involving multiplayer interactions or elements of uncertainty, such as *Chinese Checkers* [33], *Magic: The Gathering* [38], and *Scotland Yard* [23]. Monte Carlo simulations are also instrumental in the design and analysis of educational games. For instance, Fitrianawati et al. [8] leveraged Monte Carlo simulations to enumerate all possible solutions for each draw in the arithmetic card game 24. This enabled the assignment of difficulty levels to different card combinations, contributing to the mitigation of students' mathematics anxiety.

3. Presentation of the board game YETI

YETI is a collaborative board game designed for application in a higher education context, with a focus on the topic of infinite series and their direct comparison test. Infinite series play a significant role in various disciplines, including finance, physics, statistics, and engineering. Determining the convergence property of an infinite sum presents a significant challenge which can be tackled using convergence tests. The n^{th} term test for divergence is a straightforward technique stating that if the terms of an infinite sum do not approach zero, the sum diverges. Cauchy's test, also known as the root test, involves taking the n^{th} root of the absolute value of the n^{th} term, and evaluating its limit as n approaches infinity. Similarly, the application of D'Alembert's test entails calculating the limit of the ratio of the series' successive terms. The direct comparison test asserts that if an infinite sum of non-negative terms has a majorant that converges, the original sum also converges. Conversely, an infinite sum having a divergent minorant of non-negative terms indicates that the original sum diverges. The application of the direct comparison test requires an intuitive initial guess, making its usage considerably less algorithmic than the application of the other tests mentioned above. Depending on whether the examined infinite sum is believed to converge or diverge, one must find a sufficiently simple convergent majorant or divergent minorant to support this belief. The development of this intuitive approach necessitates extensive practice, the process of which can be supported by playing the board game YETI.

At the outset of the development process of YETI, it was essential to identify the primary aesthetic goals of the game to enable the use of a top-down design approach. The new tabletop board game aimed, most of all, to present a challenge for learners through its main didactic content, create a collaborative environment, offer sensory experiences through tactile and visual elements, and feature a simple yet effective narrative. A decision was made early on to design a cooperative rather than a competitive game, encouraging students with a strong grasp of the given topic to help their peers understand the course materials. Parallel to the top-down development, a bottom-up design approach was also employed, focusing on the key mechanics of the new game, with all six heuristics presented by Upton [37] thoroughly considered.

A common challenge educators face when designing didactic games is the fragility of the strategy-luck equilibrium in the game mechanics. For a game to serve as an effective learning tool, it must incorporate a substantial portion of the course materials. In addition, a degree of uncertainty is critical for maintaining the game's enjoyment factor, as Upton [37] suggests. Introducing elements of randomness, such as cards, dice, or spinners, is necessary to prevent the game from becoming a mere exercise sheet in an unusual format rather than a genuinely engaging educational experience. However, if winning is perceived to be based more on luck than on skill and comprehension, players may lose motivation to engage with the course materials, thus undermining the educational objectives of a didactic game. Therefore, although the inclusion of a card deck is pivotal in the game YETI, it must be ensured that the role of luck does not overshadow the importance of actual knowledge. Given the wide variance in potential players' skills, differentiation, achieved by developing card decks of three levels of difficulty, was key to maintaining the game's satisfaction aspect through adapting the gameplay experience to the prior knowledge of participants.

Beyond these considerations, it is essential to acknowledge that the complexity of a game strongly impacts its effectiveness as a teaching tool. Based on his extensive research on the topic, Sousa [32] states that didactic games that students are likely to play only once should have a low level of complexity. Unknown, original games necessitate debriefing and the continuous presence of facilitators to support gameplay. Given that the rules of YETI are not derived from another well-known game, additional time must be allocated for concise explanations of the game's narrative and rules before each session. Therefore, it is essential to keep the rules simple and intuitive, facilitating an immediate understanding of the game logic. Additionally, it is important to consider the time constraints and repeatability of the game. Since a university practical lesson has a duration of around 90 minutes, the gameplay must not exceed 35-40 minutes.

With these guiding principles in mind, the main components of YETI were developed, featuring an immersive narrative, an original set of rules, a versatile game mat, and three card decks of varying levels of difficulty, ranging from beginner to expert. The game encourages players to engage with the direct comparison test in a hands-on, interactive manner, promoting a deeper understanding of the subject material through repeated practice and collaborative problem-solving.

3.1. Narrative

The game's narrative revolves around a Yeti attack in a small mountain village. Players are tasked with repelling the invading beasts before they destroy the entire village. Players can gain territory by identifying valid pairs among the infinite sums on the cards in the YETI deck. A valid pair consists of a divergent sum and its divergent minorant, or a convergent sum and its convergent majorant. Within the game's narrative, these sums act as identifiers for different Yeti hunters, who must be paired into squads of two based on their fighting styles, determined by the divergence or convergence of their identifiers. If at least two squads are present in the same neighbourhood of the village, any Yeti previously based there is successfully driven away, and no further invaders dare to enter the given area. However, if two Yetis occupy the same neighbourhood, they soon engage in a territorial fight and destroy the entire village as a consequence, an unfavourable outcome which players must prevent.

3.2. Game mat

The structure of the YETI game mat, as seen in Figure 1, is clean and simple, with designated slots for Yeti or series cards. The illustrations on the mat depict a small valley village with tiny houses and meandering paths, surrounded by snowy mountain peaks. Four card slots of the same colour constitute a field, symbolising a neighbourhood within the fictional village. The relational operators between neighbouring card slots indicate the direction of relations between the pairs of infinite sums that players must respect when placing cards onto the mat. Three distinct field colours are used: dark blue indicates convergence and light blue signifies divergence, while grey card slots serve as wild cards, allowing players to play both their convergent and divergent pairs [35]. An additional rule, reinforced by the relational operators on the game mat, restricts the placement of infinite sums with equal n^{th} terms as pairs to the grey fields.

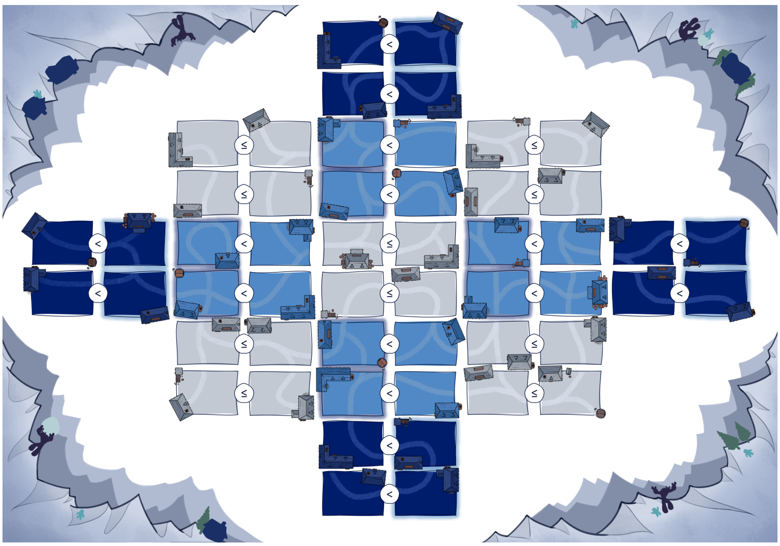

Figure 1. Game mat graphics for the board game YETI.

3.3. Set of rules

The beginning of the game YETI is marked by players strategically positioning five Yeti cards onto five different fields of their choosing on the game mat. Subsequently, the remaining deck comprising 95 cards is shuffled, and players start drawing cards. Upon encountering a Yeti card, it must be promptly laid down on any unoccupied field. If no such field is available, resulting in multiple Yeti cards being placed onto the same field, the game is lost immediately. Otherwise, drawing continues until players collectively have exactly 12 series cards in their possession, signalling the onset of the pairing phase. During the pairing phase, participants engage in discussions regarding potential card pairings, and lay down the pairs of infinite sums they identify. Successfully filling a field of four card slots results in the expulsion of any Yeti previously occupying the given field, enabling players to discard the corresponding Yeti card. The game culminates in victory if all Yeti cards are removed from the mat and more than four fields are filled, or if at least eight fields are filled. However, defeat ensues if players possess the maximum of 12 series cards but are unable to form valid pairs.

4. The tuning process of the game

The final stage of game design typically involves playtesting and tuning [14]. During the development of the board game YETI, dynamics originating from proposed sets of mechanics were continuously tested and refined using computer simulations. Through the iterative improvement of game mechanics and parameters, specific learning objectives and aesthetic goals could be reached. The tuning process consisted of two major steps: marking the exact infinite series featured in the card decks and establishing the optimal values for all numerical game parameters. In addition to these endeavours, it was imperative to address certain concerns raised by the cooperative nature of the game, such as the Alpha Player Problem.

4.1. The composition of the card decks

The development of the YETI card decks on three different levels posed a considerable challenge for multiple reasons. Primarily, the 80 infinite sums of non-negative terms featured in a deck must represent the various series types and solution methods covered in the course materials. Moreover, determining an optimal ratio of Yeti cards to series cards, achieved through extensive playtesting, is essential to align with Upton's satisfaction heuristic mentioned in Section 2. Another critical demand is guaranteeing the pairability of randomly drawn cards, increasing the likelihood of identifying pairs among the cards dealt at the beginning of each round, thus maintaining the game's dynamic flow. This requires each series card to be compatible with multiple other cards, ideally between three to five, within a deck.

To meet these criteria, a systematic methodology was employed for selecting the series included in the card decks, leading to the establishment of clusters of infinite sums with the following properties:

1. Infinite sums within the same cluster either all converge or all diverge, allowing the entire cluster to be classified as either convergent or divergent.
2. Half of the infinite sums in each cluster are simpler majorant or minorant series (depending on the cluster's convergence property).
3. Majorants or minorants within a cluster are equal, differing only in the form their n^{th} term is written in.
4. The other half of sums in a cluster are more intricate infinite series to which the direct comparison test needs to be applied after finding a suitable majorant or minorant.
5. All of the intricate series within a cluster can be paired with any majorant or minorant in the same cluster.
6. Majorants or minorants within a cluster can also be paired with each other but only on the grey fields of the game mat due to their equality.
7. Valid pairings can also be established between infinite sums in different clusters. The clustering approach only serves as a guarantee for a minimum number of possible pairings within a deck.

Figure 2 provides an illustrative example of clusters.

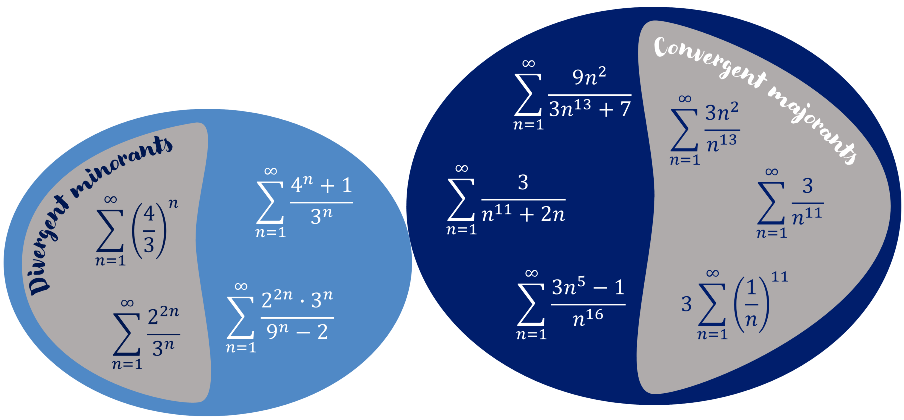

Figure 2. An illustrative example of the structure of clusters.

The deliberate construction of clusters with the specified properties ensured an adequate number of valid pairings within the game decks. Consider, for instance, a cluster of size $s \in \{2k \mid k \in \mathbb{Z}^+\}$, bearing in mind that such a cluster must always contain an even number of cards. This cluster consists of $\frac{s}{2}$ minorants/majorants and the same number of more intricate series. The intricate series are guaranteed to form valid pairs with each majorant or minorant within the cluster, while the minorants and majorants can form valid pairs with all other members of the cluster, including other minorants and majorants, except for themselves. Thus, the total number of possible pairings within a cluster is given by:

$$\frac{s}{2} \cdot \frac{s}{2} + \frac{\left(\frac{s}{2} \cdot \left(\frac{s}{2} - 1\right)\right)}{2} = \frac{3s^2}{8} - \frac{s}{4}$$

Table 1. The number of pairing possibilities within a cluster depending on cluster size.

Cluster size	Number of pairing possibilities
2 cards	1
4 cards	5
6 cards	12
8 cards	22
10 cards	35

Table 1 presents the number of guaranteed pairing possibilities within a cluster of a given size for all five cluster sizes appearing in the YETI decks. Table 2 details the distribution of clusters by size and convergence property across the various levels of YETI. Each deck, whether beginner, intermediate, or expert, contains exactly six divergent and eight convergent clusters, resulting in a total of 14 clusters. On average, each cluster comprises approximately $80/14 \approx 5.71$ cards, leading to a mean value of pairing possibilities between 5 and 12 per cluster, and, consequently, a total of roughly $5 \cdot 14 = 70$ to $12 \cdot 14 = 168$ guaranteed pairs per deck.

Table 2. The distribution of clusters by size and convergence property in the three YETI decks. The abbreviation 'div.' denotes divergent and 'conv.' denotes convergent.

Cluster size	Number of clusters					
	Beginner		Intermediate		Expert	
	Div.	Conv.	Div.	Conv.	Div.	Conv.
2 cards	2	0	2	0	2	0
4 cards	1	1	1	3	1	2
6 cards	1	5	1	2	1	3
8 cards	2	2	2	2	2	3
10 cards	0	0	0	1	0	0
all	6	8	6	8	6	8

Following the selection of the series featured in the game, abstract models of the card decks were constructed as directed graphs, representing the relationships between the infinite sums within a deck. Each infinite series in a deck, associated with a vertex in a directed graph, was assigned a unique identifier. The edges of the graphs were defined by the relationships between the n^{th} terms of the sums, pointing from the smaller n^{th} term to the larger, and from the smaller identifier to the larger in cases of equality. It is crucial to note that these models do not encompass all possible relationships between the infinite series showcased on the cards, since the primary objective of YETI is to help students build an intuition for recognising valid pairings that are easily identifiable without using a calculator. Thus, in the graphs, only these straightforward pairing possibilities are represented by edges.

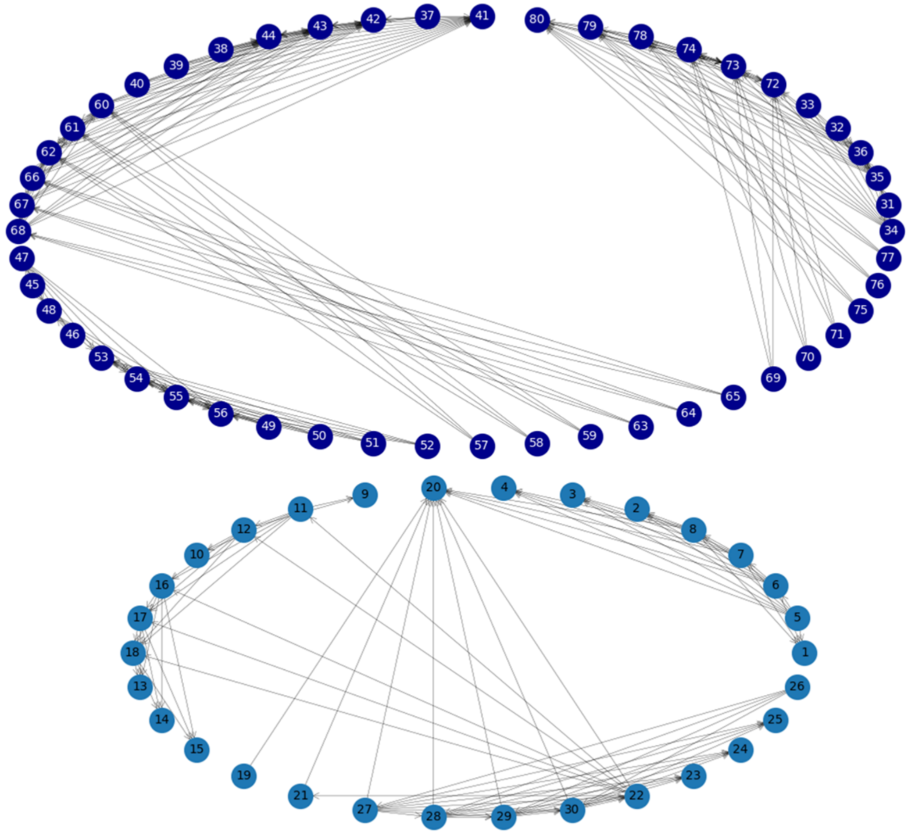

Figure 3. Directed graph $G_1 = (V_1, E_1)$ of the relations between the infinite sums in the beginner deck, limited to the relations easily noticeable to the naked eye. The vertices representing divergent sums are light blue, while the convergent ones are dark blue. $|V_1| = 50 + 30 = 80$, $|E_1| = 177 + 88 = 265$.

Initially, the abstract representations of the decks were restricted to the most easily identifiable relations. However, more advanced players might uncover additional relationships between the infinite series on the cards, including those stemming from the transitive property of inequalities. Both the graphs depicting only the most easily identifiable relations (Figure 3) and those illustrating single transitive relations (Figure 4), the recognition of which requires more practice and insight, were generated using the *NetworkX* Python package for the creation and manipulation of complex networks [22]. In the context of the subsequent game simulations, these two slightly different model types served as approximations for the knowledge of complete beginners and more experienced players.

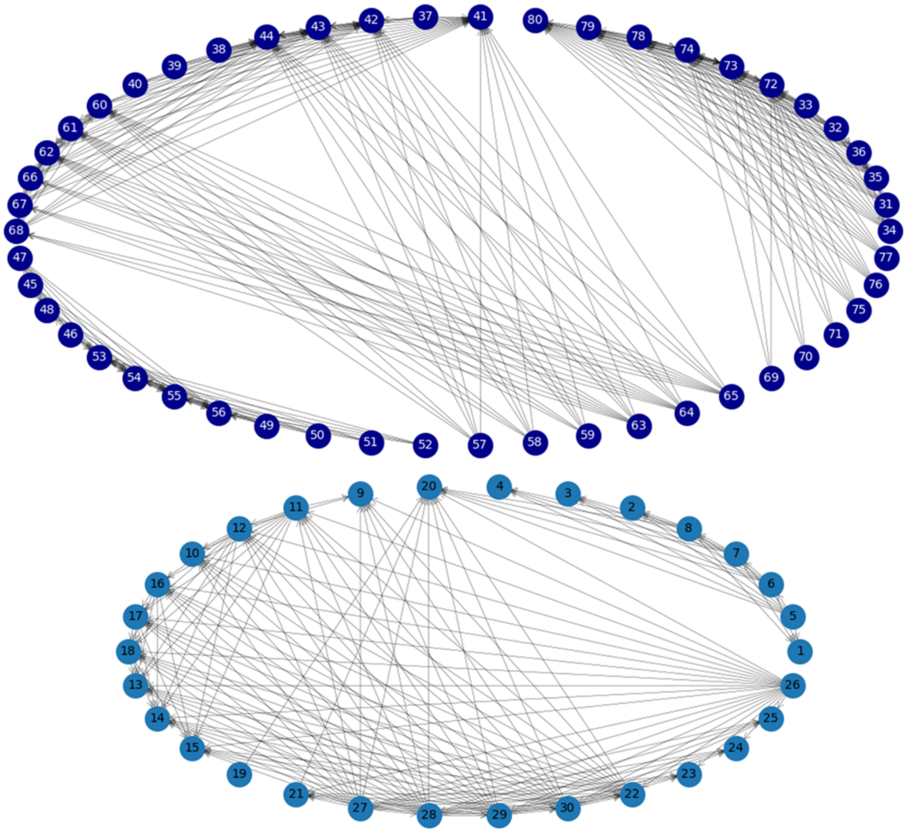

Figure 4. Directed graph $G_2 = (V_2, E_2)$ of the relations between the infinite sums in the beginner deck, including the relations formed using the transitive property of inequalities. The vertices representing divergent sums are light blue, while the convergent ones are coloured dark blue.
$|V_2| = 50 + 30 = 80, \; |E_2| = 248 + 159 = 407.$

For instance, consider the following convergent sums of non-negative terms in the beginner deck:

$$\sum_{n=1}^{\infty} \frac{6^n}{5^{2n}+3n} \quad \sum_{n=1}^{\infty} \left(\frac{3}{4}\right)^n \quad \sum_{n=1}^{\infty} \left(\frac{6}{25}\right)^n$$

For all $n \in \mathbb{Z}^+$, the easily identifiable relations between the n^{th} terms are:

$$1) \quad \frac{6^n}{5^{2n}+3n} < \frac{6^n}{5^{2n}} = \frac{6^n}{25^n} = \left(\frac{6}{25}\right)^n$$

$$2) \quad \left(\frac{6}{25}\right)^n < \left(\frac{6}{8}\right)^n = \left(\frac{3}{4}\right)^n$$

With a bit of practice, the two inequalities above can be combined forming the following relation:

$$3) \quad \frac{6^n}{5^{2n}+3n} < \left(\frac{6}{25}\right)^n < \left(\frac{3}{4}\right)^n$$

Thus, three possible pairings exist among the given infinite series:

$$P = \left\{ \begin{array}{c} \left(\sum_{n=1}^{\infty} \frac{6^n}{5^{2n}+3n}, \sum_{n=1}^{\infty} \left(\frac{6}{25}\right)^n\right), \left(\sum_{n=1}^{\infty} \left(\frac{6}{25}\right)^n, \sum_{n=1}^{\infty} \left(\frac{3}{4}\right)^n\right), \\ \left(\sum_{n=1}^{\infty} \frac{6^n}{5^{2n}+3n}, \sum_{n=1}^{\infty} \left(\frac{3}{4}\right)^n\right) \end{array} \right\}$$

4.2. Monte Carlo simulations in the design process

During the tuning process of the game YETI, Monte Carlo simulations were employed to determine the optimal values of key numerical game parameters, such as the number of Yeti cards in a deck. Considering that a deck of 100 cards can be shuffled in 100! different ways, exhaustive playtesting was deemed unfeasible, necessitating a complex, multi-parameter optimisation process, which would have been not only impractical but also highly susceptible to error if performed manually. Consequently, computer simulations were used for experimental purposes. Despite the series cards in the YETI deck forming an intricate network of interconnections, in an ideal game scenario, where player decision-making is optimised, the outcomes depend solely on the initial shuffle of the card deck, which introduces the only element of uncertainty into the game. Therefore, computer simulations focused on card shuffling were used to conduct a statistical analysis of the game, generating thousands of randomised sample game setups.

4.2.1. Optimisation targets

The primary objective of the optimisation process was to achieve a balance between the strategic and stochastic elements of the game, while minimising the occurrence of unfavourable scenarios such as premature victories, stalemates, and inevitable losses. Based on this principle and the didactic goals of the game, the following optimisation targets were defined:

1. *Coverage*: Players should be incentivised to seek both convergent and divergent pairs, primarily pairing the simpler infinite series, intended as majorants or minorants, with more intricate series.
2. *Seamless start*: There should always be easily identifiable pairs among the dealt infinite series at the beginning of the game. The deck shuffle should result in stalemates in no more than 1% of cases.
3. *Incentive to think and act*: Players should be motivated to play a minimum of 2 pairs of infinite sums on average per round.
4. *Balance of knowledge and luck*: As per Upton's insights detailed in Section 2, the uncertainty introduced by the card deck should be offset by ensuring that optimal decisions lead to victories. Experienced players should lose no more than 10-15% of their games when making optimal decisions, aligning with Upton's satisfaction heuristic.

4.2.2. Methodology

The law of large numbers asserts that as the number of trials in random experiments increases, the average of the results obtained from these trials will converge to the expected value of the underlying probability distribution. Essentially, with an infinite number of trials, empirical results will match theoretical probabilities. Guided by this principle, a large number of simulations were carried out for different potential values of the key YETI game parameters, where the relative frequency of certain designated adverse cases was recorded and analysed in order to identify an optimal setup. These simulations, implemented in Python and interfacing with a database of the infinite sums featured in the card decks, ranged from simple card shuffling to comprehensive game simulations, with algorithms replacing the mid-game decision-making of human players.

In the game simulations, two distinct decision-making models were employed: simplified and optimal moves. In each round, either the first identified pair was immediately placed on the game mat (simplified move), or the pairing process was optimised to maximise the number of cards played (optimal move). In the latter case, all potential pairs among the dealt series cards were aggregated into a set P. Next, it was determined whether the elements in each subset of P had any overlap. The optimal move for the given round was selected as the largest subset containing independent pairs of infinite series, meaning all pairs in the subset could be placed onto the game mat simultaneously. While the optimal move simulations naturally yielded more realistic outcomes, they significantly increased the program's runtime due to the exhaustive process of generating and evaluating these subsets.

For instance, suppose that the following pairs can be played in a round:

$$P = \left\{ \begin{array}{c} \left(\sum_{n=1}^{\infty}\left(\frac{6}{25}\right)^n, \sum_{n=1}^{\infty}\left(\frac{3}{4}\right)^n\right), \left(\sum_{n=1}^{\infty}\frac{6^n}{5^{2n}+3n}, \sum_{n=1}^{\infty}\left(\frac{6}{25}\right)^n\right), \\ \left(\sum_{n=1}^{\infty}\frac{3^n-1}{4^n+1}, \sum_{n=1}^{\infty}\left(\frac{3}{4}\right)^n\right), \left(\sum_{n=1}^{\infty}\frac{6^n}{5^{2n}+3n}, \sum_{n=1}^{\infty}\left(\frac{3}{4}\right)^n\right) \end{array} \right\}$$

In a simplified move, M_1 represents the set of pairs to be placed. The remaining potential pairs cannot be utilised since each infinite sum appears only once per deck.

$$M_1 = \left\{\left(\sum_{n=1}^{\infty}\left(\frac{6}{25}\right)^n, \sum_{n=1}^{\infty}\left(\frac{3}{4}\right)^n\right)\right\}$$

In contrast, the optimal move simulation, by generating all subsets of P, can produce a more favourable, larger set of pairs to be placed, denoted as M_2:

$$M_2 = \left\{\left(\sum_{n=1}^{\infty}\frac{6^n}{5^{2n}+3n}, \sum_{n=1}^{\infty}\left(\frac{6}{25}\right)^n\right), \left(\sum_{n=1}^{\infty}\frac{3^n-1}{4^n+1}, \sum_{n=1}^{\infty}\left(\frac{3}{4}\right)^n\right)\right\}$$

4.2.3. Avalanche cards

The rules of YETI create an imbalance between the fields on the game mat, with the grey card slots becoming overpowered compared to the blue ones. While blue card slots can only accommodate either convergent or divergent pairs, grey fields can house both types, effectively doubling the number of playable pairs. Additionally, pairs with identical n^{th} terms can only be placed onto grey fields. Consequently, players are incentivised to prioritise filling the grey card slots first, where even the simplest of pairs can be laid down, leading to a potential imbalance in the game's didactic content.

To address this issue, avalanche cards were introduced. An avalanche card can clear the contents of a field of a specific colour, providing a means to discard Yeti cards effectively. There are three sorts of avalanche cards, corresponding to the three primary field types: convergent, divergent, and mixed avalanches. These are distinguished by the colour of the avalanche card's borders and are only applicable to a field of the matching colour. Moreover, avalanche cards must be played during the round in which they are drawn. Therefore, to use avalanches to their advantage, players must distribute their Yeti cards across fields of different colours early in the game, as the timing and the appearance order of avalanche cards are unpredictable. All things considered, the introduction of avalanches encourages players to keep each distinct field type in the game, seeking both divergent and convergent pairs of infinite sums, as well as both simpler and more intricate pairings throughout the game, meeting the requirements of the *coverage* optimisation target.

4.2.4. The number of cards drawn per round

The number of cards dealt per round, denoted by k, is one of the most significant game parameters. In setting this parameter, the second optimisation target, *seamless start*, was considered, ensuring that players could begin their games in at least 99% of cases after drawing the first k series cards. To prevent a stalemate, it is evident that there must be at least one valid pair among the dealt cards, suggesting that a higher value of k might be reasonable. However, to avoid overwhelming players, the count of infinite sums they need to manage each round must be minimised. Consequently, the value of k must be set to the smallest possible value that meets the *seamless start* target.

To assess all potential values of k, a straightforward deck shuffling simulation was implemented. The `shuffle()` method from the random module in Python was invoked on a list comprising all the infinite sums featured in a deck. This list served as an approximation for the shuffled deck of cards. Through numerous shuffling simulations for different values of k, it was examined how frequently the initial shuffle resulted in unfavourable starting scenarios, defined as instances where a stalemate ensued at the game's onset. The shuffling experiment was conducted $N = 100{,}000$ times for each examined value of k, the results of which are detailed in Table 3, where f denotes the ratio of experiments resulting in a stalemate.

Table 3. Results of the simulations for different values of k out of $N = 100{,}000$ experiments per parameter value, where f denotes the percentage of simulations with unfavourable outcomes.

k	5	6	7	8	9	10	11	12	13
f (%)	41.85	27.41	16.34	9.14	4.85	2.32	1.07	0.47	0.18

As illustrated by Table 3, a lower value of k correlates with a higher incidence of adverse cases. For instance, when only five cards were dealt per round, the relative frequency of stalemates at the beginning of the game exceeded 40%. The lowest possible value of k where the percentage of stalemates was deemed acceptable is 12, with the ratio of adverse cases falling below 1%. Thus, the value of the parameter k has been finalised: in each round of the game YETI, players must draw enough new cards to ensure that the total number of series cards in front of them amounts to 12.

4.2.5. The number of Yeti cards per deck

To meet the third optimisation target, it was essential for YETI to encourage participants to identify at least two valid pairs of infinite series per round. The Yeti cards featured in the game deck serve as the primary incentive for this, as an excess of Yetis on the game mat can quickly lead to defeat. Evidently, a lack of Yeti cards would render the gameplay monotonous, depriving it of excitement. In contrast, an excess of Yeti cards would make victory nearly unattainable, even for the most experienced players. Therefore, to promote an engaging and motivating player experience, the appearance of one new Yeti per round was deemed ideal, inspiring the placement of two pairs of infinite sums in each round to avoid an increase in the number of Yeti cards on the game mat. Nevertheless, any scenarios requiring players to play more than three Yeti cards in quick succession were to be prevented. Hence, fine-tuning the number of Yeti cards per deck, denoted by y, became a critical step in the optimisation process.

In the shuffling simulation, the distribution of Yeti cards in a deck was modelled by adding $y-5$ Yeti elements, with the five initially placed Yeti cards substracted, to the list representing the dealer deck. The YETI game mat accommodates a total of 13 fields. To motivate players to make their pairing decisions as quickly and efficiently as possible, it was necessary to maintain the possibility of two Yetis being forced onto the same field, which constitutes a key condition for defeat. Consequently, the minimum value considered for y had to be $13+1=14$. The main optimisation goal regarding the number of Yeti cards was to minimise the frequency of scenarios where numerous consecutive Yetis appear in a deck. The upper limit for an acceptable cluster size of Yeti cards was fixed at three; any cluster exceeding this size was considered an adverse case. Thus, the probability of encountering at least one group of four or more consecutive Yeti cards in the shuffled deck was estimated as a function of the discrete parameter y.

The shuffling simulation was run $N = 100.000$ times for each examined value of y. The results are presented in Table 4, where f denotes the percentage of experiments where clusters of four or more Yeti cards were present in the shuffled deck. The table indicates that the values of y for which the ratio of unfavourable shuffle outcomes did not exceed or barely exceeded 1% fell between 14 and 16. However, further experiments were necessary to adjust the value of y, ensuring that approximately one Yeti would appear in each round.

Table 4. Results of the simulations for different values of y out of $N = 100{,}000$ experiments per parameter value, where f denotes the percentage of simulations with unfavourable outcomes.

y	14	15	16	17	18	19	20
f (%)	0.44	0.66	1.00	1.43	1.96	2.62	3.43

In addition to modelling card shuffling, a comprehensive simulation of YETI was also implemented to draw conclusions about the balance between uncertainty and predictability in the game. Two simulation variations were distinguished based on the simulated players' experience level. For modelling beginner gameplay, only the most easily identifiable relations, illustrated by Figure 3, were taken into account when pairing infinite sums. In contrast, for approximating the decisions of experienced players, pairs generated through transitivity, as seen in Figure 4, were also included. Experiments were conducted using both the simplified and the optimal move types detailed in Subsection 4.2.2.

The first task to be addressed using comprehensive game simulations was setting the number of Yeti cards in a deck. The range of possible values had previously been narrowed down to the set $\{14, 15, 16\}$ based on the deck shuffling simulations. It had also been stated that, ideally, approximately one new yeti would emerge in each round. The fluctuation of the average number of Yetis appearing per round was examined as a function of y, the count of Yeti cards in a deck. The data from simulations encompassing $N = 10{,}000$ game iterations are detailed in Table 5. The results indicate that the average value of one Yeti per round is most closely approached with $y = 16$, marking 16 as the optimal number of Yeti cards in a deck.

Table 5. The average number of Yeti cards emerging in a round as a function of y, out of $N = 10{,}000$ simulations per parameter value.

Player experience	Move type	y		
		14	15	16
Beginner	simplified	0.69	0.77	0.86
	optimal	0.72	0.81	0.90
Experienced	simplified	0.69	0.78	0.85
	optimal	0.72	0.81	0.89

4.2.6. Comprehensive game simulations

Following the establishment of key game parameter values, additional simulations were executed to examine various aspects of YETI's game dynamics. These experiments aimed to determine the total number of rounds played and pairings made in a game, the percentage of games resulting in a win, and the proportion of losses attributable to either an inability to make a move or an excess of Yeti cards on the game mat. Arithmetic means were calculated based on $N = 10{,}000$ comprehensive game simulations, summarised in Table 6.

Table 6. Sample averages for $N = 10{,}000$ comprehensive game simulations without the *reinforcement* option.

Player experience	Move type	Number of rounds	Number of pairs	Victory (%)	Loss	
					Stalemate (%)	Multiple yetis per field (%)
Beginner	simpl.	5.53	9.75	42.06	57.18	0.76
	opt.	5.46	9.94	44.77	54.55	0.68
Experienced	simpl.	5.17	12.03	71.38	27.95	0.67
	opt.	4.99	12.19	73.68	25.53	0.79

The data collected indicate that the most common cause of defeat for both novice and experienced players in the game YETI is a stalemate, an inability to make a move. Only a negligible percentage of losses results from multiple Yetis occupying a single field. On average, the game concludes after 5-6 rounds, with experienced players playing approximately two more pairs of cards than beginners. Regrettably, the ratio of games won turns out to be notably low, falling below 50% for novices and below 75% for experienced players, with a high number of losses attributable to stalemates. This suggests that even players making optimal decisions might lose a significant portion of their games, implying an imbalance between strategy and luck in the established rules and mechanics of YETI, which may negatively impact player motivation.

Table 7. Sample averages for $N = 10{,}000$ comprehensive game simulations integrating the *reinforcement* option.

Player experience	Move type	Number of rounds	Number of pairs	Victory (%)	Loss	
					Stalemate (%)	Multiple yetis per field (%)
Beginner	simpl.	7.31	11.88	69.23	29.71	1.06
	opt.	7.12	11.96	70.83	27.99	1.18
Experienced	simpl.	6.02	13.15	89.89	9.32	0.79
	opt.	5.69	13.28	91.24	7.74	1.02

To mitigate this problem, a new optional action named reinforcement was added to the game mechanics. Once per game, this rule allows players to replace three of their series cards with new cards from the dealer deck. Should a Yeti card be dealt during this process, it can be shuffled back into the deck, and another card can be drawn in its place. The simulation results for $N = 10{,}000$ games integrating the reinforcement option are detailed in Table 7. In the simulation of the reinforcement option, the series cards selected for discarding are chosen randomly without any optimisation. Still, its integration results in a considerable increase in both the number of rounds and the number of pairs played. More significantly, a shift in the victory rate reflects a substantial improvement: leveraging the reinforcement option, beginners, when making optimal decisions, can win approximately 70% of their games, while experienced players can benefit from a victory rate of around 90%.

It is evident that the YETI game variant with the parameter values $k = 12$ and $y = 16$, along with the addition of avalanche cards and the reinforcement option, fully meets the optimisation targets outlined in Subsection 4.2.1. However, it is important to emphasise that the parameter optimisation process was focused solely on game mechanics. To validate the effectiveness and assess the educational content of the game, further experiments with human participants, who are subject to errors and learning curves, are necessary. Additionally, potential issues arising from the collaborative nature of the game warrant further investigation.

4.3. Cooperativity variants of the game

Drawing from their research on mathematical board games, Nurnberger-Haag, Wernet, Benjamin [25] propose that an effective didactic board game should encompass competitiveness and asynchronicity, integrate elements of luck, strategy, and mathematical knowledge, and be suitable for both introducing and revising a specific topic. However, their study exclusively examines competitive games, and the authors themselves acknowledge that their recommendation is not definitive regarding the question of competition versus cooperation. Despite this limitation, YETI, with an added feature supporting asynchronous play, aligns with the enumerated recommendations. The proposed modification can be implemented following the drawing phase, when players cooperatively identify pairs of infinite series from the pool of dealt cards. During this phase, each player is encouraged to choose a series card and formulate a hypothesis regarding the convergence property of the infinite sum showcased on it. If the others agree with the hypothesis, the selected card is set aside into a group labelled as “convergent” or “divergent”. In contrast, disagreements result in the chosen card remaining in the center, indicating that its convergence property remains undetermined. This individual guessing exercise not only introduces an asynchronous element into the game, but also simplifies the pairing process by narrowing down the pool of possible combinations, as only infinite sums with matching convergence properties can be paired up in the game.

The collaborative aspect of the game YETI prompts several key considerations. A common issue in cooperative games is referred to as the Alpha Player Problem, where a dominant player dictates the group’s actions and strategies, reducing the input and control of others, thus diminishing their engagement and enjoyment. YETI addresses this by allowing players to take turns hypothesising the convergence of the dealt series at the beginning of each round, ensuring active participation from all. Additionally, all pairing decisions must be explained, providing an opportunity for less experienced players to learn from their more skilled counterparts. Since YETI is recommended for supervised play, educators also have the opportunity to address the Alpha Player Problem directly by either reassigning dominant players to more suitable groups or creating new groups equipped with more challenging decks.

The game’s structure is also adaptable to mitigate the Alpha Player Problem. In collaborative game design, the most widely implemented solutions to prevent a single player’s dominance include rotating leadership roles, realising exceptionally quick gameplay, introducing hidden information, or incorporating a traitor mechanic. The latter two can be applied to the game YETI as well. The traitor can be narratively presented as a spy from a rival village, aiming to mislead players into making at least five pairing mistakes. If successful, the traitor wins the game, and the others lose. Naturally, the impact of the traitor role requires thorough evaluation regarding its effect on players, its impact on the learning process, and the potential confusion it may cause for less experienced players. Alternatively, hidden information can be introduced by dealing each player two hidden series cards at the beginning of the game. In this case, players have the option to exchange one card with another player each round, gradually becoming acquainted with all hidden cards. If a player obtains pairable infinite sums as hidden cards, they can place them on the game mat at the beginning of the round. This modification prevents the dominance of alpha players, maintaining a balanced and engaging gameplay experience for each participant.

5. Conclusion

This paper presented the intricate process of fine-tuning the didactic board game YETI, which aims to familiarise students with the direct comparison test of infinite series. Key aspects addressed by the tuning process include building a systematic approach to marking the infinite sums featured in the card decks, tackling concerns related to cooperativity, and optimising the key numerical parameters of the game. In this endeavour, important targets such as balancing strategy and luck, achieving seamless starts, and encouraging player engagement were reached. Monte Carlo simulations were employed to resolve potential gameplay issues such as early victories, stalemates, and guaranteed losses. Additionally, new gameplay features like the reinforcement option and avalanche cards were introduced as part of the optimisation process to maintain a challenging yet enjoyable player experience.

In board game research, Monte Carlo simulations are a well-established tool, primarily used for developing AI-driven game agents and simulating opponents via Monte Carlo Tree Search (MCTS) [1, 9, 23, 33, 38]. While there have been attempts to leverage these simulations for fine-tuning mechanics or exploring game configurations, such uses have largely been limited to computer games [12, 13, 15, 16]. It is also important to note that, so far, the Monte Carlo method has predominantly been used as a technique for analysing adversarial, deterministic, turn-based games [1, 5, 9]. Using YETI as a case study, our research illustrates how Monte Carlo simulations can contribute to optimising game mechanics and achieving balanced gameplay in the context of a non-deterministic, non-digital, cooperative didactic board game.

The vast number of possible scenarios in game design makes manual testing impractical and time-consuming, covering only a small area of the game space. This is where Monte Carlo simulations prove valuable, bridging the gap between the invention of mechanics and the assessment of the resulting dynamics. The utilisation of simulations in the case of YETI enabled a thorough examination of various game parameters, which significantly expedited the design process by reducing the reliance on time-consuming manual testing. This efficiency allowed for a comprehensive analysis of potential gameplay scenarios within a shorter time frame, ensuring that the design process was concluded promptly and effectively. Nevertheless, it is essential to acknowledge the limitations of such simulations. While they are highly effective for setting game parameter values, they are not a substitute for real-world testing. The behaviour of virtual players in simulations does not accurately reflect the playing style, learning curve, and strategic diversity of human players. Therefore, further testing involving human participants is necessary to validate the effectiveness and educational value of the game.

As evidenced by our research, tuning is an indispensable phase of didactic board game design, where the finer details of the game are established to achieve the intended player experience. This process can be approached in various ways, but it consistently demands rigorous, critical thinking, a problem-solving mindset, and a systematic approach. Monte Carlo simulations are a powerful tool in didactic game design, particularly for games where the structure can be accurately simulated. The successful application of these simulations in the development of the board game YETI highlights their potential to facilitate the game design process and provide well-balanced, engaging gameplay.

References

[1] B. Arneson, R. B. Hayward, P. Henderson: *Monte Carlo Tree Search in Hex*, IEEE Transactions on Computational Intelligence and AI in Games 2.4 (2010), pp. 251–258, doi: 10.1109/TCIAIG.2010.2067212.

[2] S. Bakkes, C. T. Tan, Y. Pisan: *Personalised gaming: a motivation and overview of literature*, in: Proceedings of The 8th Australasian Conference on Interactive Entertainment: Playing the System, IE '12, Auckland, New Zealand: Association for Computing Machinery, 2012, doi: 10.1145/2336727.2336731.

[3] C. Balakrishna: *The Impact of In-Classroom Non-Digital Game-Based Learning Activities on Students Transitioning to Higher Education*, Education Sciences 13.4, 328 (2023), doi: 10.3390/educsci13040328.

[4] C. Benedikte Søgaard Hansen, T. Bjørner: *Designing an Educational Game: Design Principles from a Holistic Perspective*, The International Journal of Learning: Annual Review 17.10 (2011), pp. 279–290, doi: 10.18848/1447-9494/cgp/v17i10/47275.

[5] C. Browne, F. Maire: *Evolutionary Game Design*, IEEE Transactions on Computational Intelligence and AI in Games 2.1 (2010), pp. 1–16, doi: 10.1109/TCIAIG.2010.2041928.

[6] G. Chaslot, S. Bakkes, I. Szita, P. Spronck: *Monte-Carlo Tree Search: A New Framework for Game AI*, Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment 4.1 (2021), pp. 216–217, doi: 10.1609/aiide.v4i1.18700.

[7] M. Dudás, S. Lengyelné Szilágyi, I. Piller: *Card Deck Designer Software for the Mathematical Game Called Ékkővadászok*, Gradus 6.4 (2019), pp. 17–27.

[8] M. Fitrianawati, Z. Aliansyah, N. R. N. Peni, I. W. Farid, L. Hakim: *Monte Carlo method at the 24 game and its application for mathematics education*, Journal of Honai Math 5.2 (2022), pp. 83–94, doi: 10.30862/jhm.v5i2.250.

[9] S. Gelly, L. Kocsis, M. Schoenauer, M. Sebag, D. Silver, C. Szepesvári, O. Teytaud: *The grand challenge of computer Go: Monte Carlo tree search and extensions*, Commun. ACM 55.3 (2012), pp. 106–113, doi: 10.1145/2093548.2093574.

[10] R. L. Harrison: *Introduction to Monte Carlo Simulation*, AIP Conference Proceedings 1204.1 (2010), pp. 17–21, doi: 10.1063/1.3295638.

[11] M. Hernandez-de-Menendez, C. A. Escobar Díaz, R. Morales-Menendez: *Educational experiences with Generation Z*, International Journal on Interactive Design and Manufacturing (IJIDeM) 14.3 (2020), pp. 847–859, doi: 10.1007/s12008-020-00674-9.

[12] C. Holmgård, A. Liapis, J. Togelius, G. Yannakakis: *Monte-Carlo Tree Search for Persona Based Player Modeling*, Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment 11.5 (2021), pp. 8–14, doi: 10.1609/aiide.v11i5.12849.

[13] B. Horn, J. A. Miller, G. Smith, S. Cooper: *A Monte Carlo approach to skill-based automated playtesting*, Proceedings of the 14th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (2018).

[14] R. Hunicke, M. Leblanc, R. Zubek: *MDA: A Formal Approach to Game Design and Game Research*, AAAI Workshop - Technical Report 1 (2004).

[15] A. Isaksen, D. Gopstein, J. Togelius, A. Nealen: *Discovering Unique Game Variants*, in: Computational Creativity and Games Workshop at the 2015 International Conference on Computational Creativity, 2015.

[16] A. Isaksen, D. Gopstein, J. Togelius, A. Nealen: *Exploring Game Space of Minimal Action Games via Parameter Tuning and Survival Analysis*, IEEE Transactions on Games 10.2 (2018), pp. 182–194, doi: 10.1109/TCIAIG.2017.2750181.

[17] A. Jaffe, A. Miller, E. Andersen, Y.-E. Liu, A. Karlin, Z. Popovic: *Evaluating Competitive Game Balance with Restricted Play*, Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment 8.1 (2021), pp. 26–31, doi: 10.1609/aiide.v8i1.12513.

[18] R. Khoshkangini, G. Valetto, A. Marconi, M. Pistore: *Automatic generation and recommendation of personalized challenges for gamification*, User Modeling and User-Adapted Interaction 31.1 (2020), pp. 1–34, doi: 10.1007/s11257-019-09255-2.

[19] A. Liapis, C. Holmgård, G. N. Yannakakis, J. Togelius: *Procedural Personas as Critics for Dungeon Generation*, in: Applications of Evolutionary Computation, ed. by A. M. Mora, G. Squillero, Cham: Springer International Publishing, 2015, pp. 331–343.

[20] K. Moore, C. Jones, R. S. Frazier: *Engineering Education For Generation Z*, American Journal of Engineering Education (AJEE) 8.2 (2017), pp. 111–126, doi: 10.19030/ajee.v8i2.10067.

[21] M. Nelson: *Game Metrics Without Players: Strategies for Understanding Game Artifacts*, Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment 7.3 (2011), pp. 19–24, doi: 10.1609/aiide.v7i3.12479.

[22] *NetworkX-NetworkX documentation*, https://networkx.org/ [Accessed: 13 Jun 2024].

[23] P. Nijssen, M. H. M. Winands: *Monte Carlo Tree Search for the Hide-and-Seek Game Scotland Yard*, IEEE Transactions on Computational Intelligence and AI in Games 4.4 (2012), pp. 282–294, doi: 10.1109/TCIAIG.2012.2210424.

[24] T. Nummenmaa, J. Kuittinen, J. Holopainen: *Simulation as a game design tool*, in: Proceedings of the International Conference on Advances in Computer Entertainment Technology, ACE '09, Athens, Greece: Association for Computing Machinery, 2009, pp. 232–239, doi: 10.1145/1690388.1690427.

[25] J. Nurnberger-Haag, J. L. Wernet, J. I. Benjamin: *Gameplay in Perspective: Applications of a Conceptual Framework to Analyze Features of Mathematics Classroom Games in Consideration of Students' Experiences*, International Journal of Education in Mathematics, Science and Technology 11.1 (2022), pp. 267–303, doi: 10.46328/ijemst.2328.

[26] S. Raychaudhuri: *Introduction to Monte Carlo simulation*, in: Proceedings of the 2008 Winter Simulation Conference, Miami, FL, USA: IEEE, 2008, pp. 91–100, doi: 10.1109/WSC.2008.4736059.

[27] J. Ross, A. Marshak: *Monte Carlo Methods*, in: Photon-Vegetation Interactions: Applications in Optical Remote Sensing and Plant Ecology, ed. by R. B. Myneni, J. Ross, Berlin, Heidelberg: Springer, 1991, pp. 441–467, doi: 10.1007/978-3-642-75389-3_14.

[28] R. Y. Rubinstein, D. P. Kroese: *Simulation of Discrete-Event Systems*, in: Simulation and the Monte Carlo Method, Hoboken, NJ, USA: John Wiley & Sons, Inc., 2016, chap. 3, pp. 91–106, doi: 10.1002/9781118631980.ch3.

[29] J. Schell: *The Art of Game Design: A Book of Lenses*, 3rd, Boca Raton, FL: CRC Press, 2019, doi: 10.1201/b22101.

[30] C. Seemiller, M. Grace: *Generation Z goes to college*, San Francisco, CA: Jossey-Bass, 2016.

[31] C. Sousa, S. Rye, M. Sousa, P. J. Torres, C. Perim, S. A. Mansuklal, F. Ennami: *Playing at the School Table: Systematic Literature Review of Board, Tabletop, and other analog game-based learning approaches*, Frontiers in Psychology 14, 1160591 (2023), doi: 10.3389/fpsyg.2023.1160591.

[32] M. Sousa: *Mastering Modern Board Game Design to build new learning experiences: The MBGTOTEACH framework*, International Journal of Games and Social Impact 1.1 (2013), pp. 68–93, doi: 10.24140/ijgsi.v1.n1.04.

[33] N. R. Sturtevant: *An Analysis of UCT in Multi-player Games*, in: Computers and Games, ed. by H. J. van den Herik, X. Xu, Z. Ma, M. H. M. Winands, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 37–49.

[34] S. Szilágyi, A. Körei: *Using a Math Card Game in Several Ways for Teaching the Concept of Limit*, in: Mobility for Smart Cities and Regional Development - Challenges for Higher Education, ed. by M. E. Auer, H. Hortsch, O. Michler, T. Köhler, Cham: Springer International Publishing, 2022, pp. 865–877, doi: 10.1007/978-3-030-93904-5_85.

[35] S. Szilágyi, E. Palencsár: *Board Games in Mathematics Education: Presentation of the PDCA-based Graphic Design Process of the YETI Didactic Framework*, Gradus 10.2 (2023), doi: 10.47833/2023.2.csc.002.

[36] J. Togelius, G. N. Yannakakis, K. O. Stanley, C. Browne: *Search-Based Procedural Content Generation*, in: Applications of Evolutionary Computation, ed. by C. Di Chio, S. Cagnoni, C. Cotta, M. Ebner, A. Ekárt, A. I. Esparcia-Alcazar, C.-K. Goh, J. J. Merelo, F. Neri, M. Preuss, J. Togelius, G. N. Yannakakis, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 141–150.

[37] B. Upton: *The Aesthetic of Play*, Cambridge, MA: MIT Press, 2015, doi: 10.7551/mitpress/9251.001.0001.

[38] C. D. Ward, P. I. Cowling: *Monte Carlo search applied to card selection in Magic: The Gathering*, in: 2009 IEEE Symposium on Computational Intelligence and Games, 2009, pp. 9–16, doi: 10.1109/CIG.2009.5286501.

[39] M. H. M. Winands: *Monte-Carlo Tree Search in Board Games*, in: Handbook of Digital Games and Entertainment Technologies, ed. by R. Nakatsu, M. Rauterberg, P. Ciancarini, Singapore: Springer Singapore, 2017, pp. 47–76, doi: 10.1007/978-981-4560-50-4_27.

[40] T. Zhang, J. Liu, Y. Shi: *Enhancing collaboration in tabletop board game*, in: Proceedings of the 10th Asia Pacific Conference on Computer Human Interaction, Matsue-city, Shimane, Japan: Association for Computing Machinery, 2012, pp. 7–10, doi: 10.1145/2350046.2350050.

[41] R. Zubek: *Elements of Game Design*, Cambridge, MA: MIT Press, 2020.