Procedural Game Level Design to Trigger Spatial Exploration

Pedro Acevedo
Purdue University
West Lafayette, U.S.A.
paceved@purdue.edu

Minsoo Choi
Purdue University
West Lafayette, U.S.A.
choi714@purdue.edu

Huimin Liu
Purdue University
West Lafayette, U.S.A.
liu2833@purdue.edu

Dominic Kao
Purdue University
West Lafayette, U.S.A.
kaod@purdue.edu

Christos Mousas
Purdue University
West Lafayette, U.S.A.
cmousas@purdue.edu

Figure 1: Example stills of the open-world game level that we synthesized based on our proposed procedural spatial exploration level design method.

ABSTRACT

Synthesizing game levels that evoke players' curiosity, driving them to explore different level parts, is time-consuming and tedious. Typically, game level designers manually perform this synthesis using trial and error. In this paper, we propose a method with which to replace this manual, time-consuming process. We benefited from recent work that had proposed game level design patterns to evoke curiosity, and we propose an approach to automatically synthesizing game levels in order to encourage players to pursue designer-specified exploration goals. We started by creating a dataset of level assets, based on the four design patterns that evoke curiosity-driven exploration in games (*reaching extreme points*, *resolving visual obstructions*, *out-of-place objects*, and *understanding spatial connections*). We annotated the assets in our dataset with spatial exploration measurements (the time players took to explore an asset over their total time spent in the game level). We then formulated game level design as an optimization problem, encoding both spatial exploration (mean spatial exploration, spatial exploration variance, and spatial exploration distribution) and game level design (occupied area, adjacent penalty, and height distribution) decisions. Then, we solved this problem by implementing a reversible-jump Markov chain Monte Carlo method. We demonstrate our method's ability to synthesize game level variations with different spatial exploration and level design decisions. Finally, a user study showed that our approach can automatically synthesize game levels, encouraging a certain amount of spatial exploration by players.

FDG '22, September 5–8, 2022, Athens, Greece

ACM ISBN 978-1-4503-9795-7/22/09.
https://doi.org/10.1145/3555858.3563272

CCS CONCEPTS

• Software and its engineering → Interactive games; **• Applied computing** → *Computer games*.

KEYWORDS

spatial exploration, curiosity, game level, level design, procedural content generation

ACM Reference Format:
Pedro Acevedo, Minsoo Choi, Huimin Liu, Dominic Kao, and Christos Mousas. 2022. Procedural Game Level Design to Trigger Spatial Exploration. In *FDG '22: Proceedings of the 17th International Conference on the Foundations of Digital Games (FDG '22), September 5–8, 2022, Athens, Greece.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3555858.3563272

1 INTRODUCTION

When designing a game level, a designer should encounter various factors that impact players' gameplay behavior, including players' "curiosity" about exploring the level and its content [28, 29]. Curiosity, from a psychological perspective, can be regarded as uncertainty and preferences regarding "information gaps" between the known and unknown [29]. In games, *curiosity* denotes a player's interest in confronting uncertainty and tolerating information gaps or a lack of context. This interest enables the map exploration process.

Due to insufficient knowledge regarding game level patterns that trigger curiosity, game level designers use their intuition to design levels that encourage this behavior among players [6, 22, 37]. A recent study by Gómez-Maureira and Kniestedt [8] analyzed the design of games that players associate with the five dimensions of curiosity (*joyous exploration*, *deprivation sensitivity*, *stress tolerance*, *social curiosity*, and *thrill seeking*) [19] and identified design patterns commonly used in games to encourage spatial exploration. Then, based on the patterns they identified, Gómez-Maureira et al. [9] empirically studied how such level design patterns influence spatial exploration in a 3D, open-world game. Such studies provide an evidence-based explanation of what game level design patterns or components evoke a desire to explore and unlock the potential for procedural generation to induce spatial exploration in some game parts.

However, while procedural game level design's potential to encourage the pursuit of a certain spatial exploration target is appealing, designing such levels remains tedious, time-consuming, and challenging—heavily relying on a game level designer's individual perception of this concept. Although some game developers incorporate narrative elements that enable players to pursue the activities they enjoy the most [11], To et al. [45] have shown that letting players encounter game content as the result of their curiosity creates more memorable and enjoyable game experiences.

Inspired by research on procedural game level design [5, 18, 20, 36] and, more specifically, methods favoring experience-driven procedural content generation [35, 40, 53], as well as curiosity in games [6, 8, 9, 22, 37, 45], we propose an optimization-based approach to procedurally generating game levels that encourage spatial exploration. Our method automatically synthesizes open-world game levels (see Figure 1 for an example of a synthesized open-world game level) that encourage players to pursue designer-specified spatial exploration patterns. We formulated spatial-exploration based level design as an optimization problem; therefore, our system automatically and quickly synthesizes game levels, balancing different design considerations. Thus, game level designers can use these synthesized levels as a basis for further refinement.

To synthesize game levels for which a designer can control both spatial exploration and level design, we formulated a method as a set of cost terms that encode both spatial exploration (mean spatial exploration, spatial exploration variance, and spatial exploration distribution) and game level design (occupied area, adjacent penalty, and height distribution) decisions. We assigned all cost terms to a total cost function, which we solved using a reversible-jump Markov chain Monte Carlo method [10]. Based on our approach, we expect that game level designers can easily specify the spatial exploration sought from players when completing a synthesized level, as we explicitly quantified such design decisions as cost terms in our method.

In addition to presenting our method in this paper, we also conducted a user study to evaluate our approach. Specifically, we explored how study participants explored game levels with different spatial exploration targets (low, medium, and high spatial exploration). We found that our system could synthesize game levels that influenced participants' spatial exploration behavior.

The remainder of this paper is organized into the following sections. In Section 2, we describe related work on procedural game level design, and curiosity and spatial exploration in games. In Section 3, we present our preliminary remarks on the annotation and representation of our level assets. In Section 4, we present our formulation and optimization of spatial exploration in game level design. In Section 5, we describe our study evaluating procedurally generated game levels with different spatial exploration targets, and we discuss our findings and limitations. Finally, we draw conclusions from our project and present avenues for future work in Section 6.

2 RELATED WORK

Procedural Content Generation. *Procedural content generation* refers to methods used to create content for games algorithmically, as opposed to manually. Procedural content generation combines human-designed assets and algorithms with computer-generated randomness. Such techniques have been used in the games industry for nearly 40 years. The academic community has actively focused on such techniques for over a decade. Procedural generation methods have been applied to game design and development, from 2D platform games [5] to first-person shooting games [3] and, more recently, to exergames.

Unsurprisingly, given procedural generation techniques' extensive history of use in the industry, many algorithmic methods have been developed and used, including generative grammars, constraint solving, and design space searches [38]. Procedural content generation provides two main game design advantages: (1) unlimited content and variation for each play session and (2) the ability to create unique experiences in a game. For unlimited content and variation, we found that games such as *Minecraft*[1] use procedural content generation to create virtual worlds, while *Fortnite Battle Royale*[2] uses randomness and content variation in-game. However, the scope of formal research on procedural content generation methods is somewhat limited. In most cases, game developers who use procedural content generation do not disclose their practices or source code. Previous research has shown that these characteristics are essential to ensuring a game's longevity and success [31, 51].

Researchers have developed numerous procedural content generation methods that address a wide range of problems [47]. Togelius et al. [48] defined various distinctions between different procedural content generation approaches. Our project focuses on search-based procedural generation through optimization [48] and experience-driven procedural design [53], as we allow the procedural generation process to be guided through the evaluation of system-proposed designs. According to previously published work, such evaluations can be based on game developers' design decisions [2] or players [12] through the implementation of various computational models of preference [25, 27], game simulations [1, 46], or cost functions [26, 42].

In most cases of procedural game content generation, developers mainly focus on a game level itself. According to previously published work [49], high randomness and variation across game levels often provide unique experiences since, during gameplay, various unexpected interactions between the player and the game can occur. However, procedural content generation is not limited to playable content, such as game levels. It has been used to allow players to create their own story evolution [43, 44], providing players a more customized interaction with games. Such approaches allow for a wide range of actions, interactions, and strategies that a game developer might not have planned or expected [43]. Moreover, such methods have also been used to design various game assets.

[1] https://en.wikipedia.org/wiki/Minecraft
[2] https://en.wikipedia.org/wiki/Fortnite_Battle_Royale

For example, *SpeedTree*[3] software provides natural-looking trees, which are common assets in games.

Numerous researchers have published papers on synthesizing virtual worlds for games. Shi et al. [39] and Jenning-Teats et al. [16] developed a rules-based approach that adjusts game difficulty during runtime, based on a player's performance. In another example, Hooshyar et al. [14] developed a data-driven method that considered students' skills in developing a customized educational game. Xie et al. [51] procedurally generated game levels for training programs, and Liu et al. [32] synthesized racket sports exergames to encourage the pursuit of a user-specified training goal. For more details regarding procedural content creation in games, see the survey paper by Hendrikx et al. [13]. Similar to Xie et al. [51] and Liu et al. [32], we study the procedural generation of game levels by combining game level design patterns that encourage curiosity and game level design decisions. In contrast to the aforementioned papers, our work focuses on a player's spatial exploration in procedural game generation.

Curiosity in Games. Researchers consider games and virtual worlds an ideal medium for studying player curiosity; however, the importance of *curiosity* is not yet evident [9, 52]. Researchers have tried to approach player curiosity from different perspectives. Costikyan [6] defined *curiosity* as a player's motivation to engage in a game, Klimmt [22] described it as the reason why players choose to play games, and Schell [37] described it as the way game designers can make players question themselves about a game's design. In this paper, we consider players' curiosity to be their desire to explore a game level and, more specifically, a level's assets (each asset belongs to a different level design pattern). According to To et al. [45], spatial exploration in a game or virtual world can result from player actions or conceptually by resolving knowledge gaps. Phan et al. [34] also described spatial exploration as a form of creative expression that contributes to a player's satisfaction with a game.

The current study aims to use knowledge from prior research on game level design patterns that evoke curiosity in a procedural game level design framework. Researchers have previously studied level design patterns [15, 21, 41]. Although various approaches can be applied to the game design process, they do not all successfully evoke curiosity among players. Moreover, little research has examined using such design patterns in procedural design. To et al. [45] provided generalizable game design guidelines by exploring how designers can influence players' curiosity through the curiosity model by Kreitler et al. [23], which distinguishes between different curiosity triggers. These guidelines provide designers with various possible design decisions to induce player curiosity. To further explain how designers can induce player curiosity, Gómez-Maureira and Kniestedt [8] analyzed game designs using the five dimensions of the curiosity scale [19]. They identified the design patterns in games that evoke spatial exploration and social interaction. In their follow-up empirical study, Gómez-Maureira et al. [9] showed how the game level design patterns they identified influence curiosity-driven exploration in a 3D, open-world game.

Through the current project, we aim to complement previous work on game level design patterns that evoke curiosity-driven exploration by implementing a method that encounters such game level design patterns and procedurally generating game levels that encourage certain spatial exploration behavior among players. We consider our project as an initial step in applying the theoretical knowledge obtained by To et al. [45], Gómez-Maureira and Kniestedt [8], and Gómez-Maureira et al. [9] to develop a method of procedurally generating game levels that trigger a certain curiosity behavior (i.e., spatial exploration) among players. Our results could provide the research community with valuable insights regarding experience-driven design for spatial exploration; therefore, researchers could build on our results to design more advanced approaches to evoke curiosity-driven exploration in games.

[3]https://store.speedtree.com/

3 PRELIMINARY REMARKS

As a preliminary step of our research, we created a dataset of game level assets that we later annotated, based on spatial exploration. In the following subsections, we describe our methodology for preparing, annotating, and representing these assets.

3.1 Game Level Assets

We started our project by defining a list of assets (or game level objects) that can attract a player's attention during open-world level exploration. Before creating a dataset of level assets, we considered the four design patterns for curiosity to determine a player's spatial exploration, as defined by Gómez-Maureira and Kniestedt [8]:

- **Reaching Extreme Points (EXP):** Games that encourage exploration often feature locations considerably higher than the rest of the game environment.
- **Resolving Visual Obstructions (OBS):** Parts of a game environment can be deliberately obscured to motivate exploration.
- **Out-of-Place (OOP):** Out-of-place elements are game objects that stand out in the context in which they are placed.
- **Understanding Spatial Connections (SPC):** Games that allow players to navigate through an environment might feature complex, interconnected paths.

Based on these four categories, we created a dataset of 32 assets (six for EXP, seven for OBS, 12 for OOP, and seven for SPC). Each corresponds to a different design pattern. We provide images and additional descriptions of all the game level assets we used as supplementary materials.

3.2 Game Level Asset Annotation

During our annotation step, we annotated all 32 game assets. We assigned spatial exploration values to characterize each game asset. These spatial exploration values correspond to players' total time spent exploring each asset. We developed the following approaches to collecting and assigning such data to each game level asset. First, we created short game levels in the Unity game engine, where we assigned a single asset to each game level. Each game level used an 8×8 grid size. Each grid cell was 25×25 Unity units in size. Depending on the (grid) size that characterized each asset, we placed each game level asset at the center of an 8×8 grid or the upper-right cell (the [5, 5] cell; [0, 0] is the bottom left corner of the 2D grid). We chose an 8×8 grid since we realized, during a preliminary annotation process, that this grid size provides sufficient space for a player to move around and explore a game level while (given the use of only one asset) providing the feeling of an open-world level. We show a few single-asset game levels used in our annotation process in Figure 2.

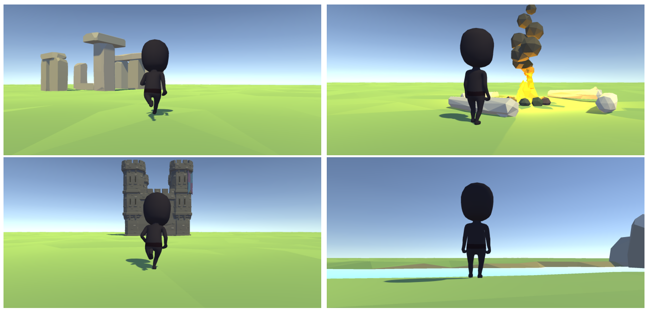


Figure 2: Example single-asset game levels used in our annotation process.

After designing all 32 game levels and receiving approval from our university's Institutional Review Board (IRB) for this part of our preliminary research, we asked both undergraduate and graduate students from our department to volunteer to play all of our game levels. We collected spatial exploration data: how much time (in seconds) a player spent inside each cell of the examined asset over the total time they spent in the level. We considered the spatial exploration of each asset by cell, rather than treating each level as a single entity, because we noticed that some parts of an asset might trigger more spatial exploration behavior among participants than other parts of the same asset.

In total, we recruited 13 participants (eight male, age: $M = 26.87$, $SD = 4.15$; five female, age: $M = 23.60$, $SD = 2.70$). Three participants indicated that they spent less than an hour per week playing games, while four played one to two hours per week, two played two to five hours per week, two played five to 10 hours per week, and two played more than 10 hours per week. Two participants characterized themselves as *novice players*, five as *casual players*, and six as *core players*. After screening our data, we noticed that the data for one of our participants had not been recorded properly; therefore, we excluded one participant and annotated our game level assets using data provided by only 12 participants. Based on previously published work [32, 51], and given the variability among participants, we considered 12 participants sufficient to provide reliable data with which to annotate our game level assets.

For the annotation process, we asked participants to play all game levels. Each game level lasted 100 seconds (a counter started once a player began a level and froze after 100 seconds). All game levels were played according to the Latin square [50] balancing method. Based on this method, for each asset, we computed the average time that all participants spent inside each cell over their total gameplay time (100 seconds); therefore, for each cell of each asset, we obtained a spatial exploration value $\in [0, 1]$. Further details about spatial exploration for each asset are provided in the supplementary material.

3.3 Game Level Asset Representation

The final step of our preliminary research was representing each game level asset. Because our procedural game level design method considers spatial exploration and some level design characteristics, we represented each game level asset a_i as $a_i = \langle \mathcal{E}(a_i), \mathcal{H}(a_i) \rangle$. $\mathcal{E}(a_i)$ denotes the spatial exploration value of each grid cell of asset a_i, computed based on the methodology mentioned in Section 3.2, represented as $\mathcal{E}(a_i) = [\varepsilon_{1,1}(a_i), ..., \varepsilon_{y,z}(a_i)]$. $\mathcal{H}(a_i)$ is a 2D array that encodes an asset's height information. Specifically, each cell of $\mathcal{H}(a_i)$ represents the highest point (height information) for asset a_i, represented as $\mathcal{H}(a_i) = [h_{1,1}(a_i), ..., h_{y,z}(a_i)]$. For representation efficiency, we assigned height values as the normalized ($\in [0, 1]$) height computed through processing all the game level assets in our dataset. (y, z) denotes the size of an asset in terms of grid cells, and the grid $\mathcal{E}(a_i)$ is equal to $\mathcal{H}(a_i)$.

4 PROBLEM FORMULATION AND OPTIMIZATION

Let L ($m \times n$ in size) denote the whole grid map we used to synthesize a game level and information about the level's configuration. We represent a level (grid map) as a 2D array: $L = [l_{1,1}, ..., l_{m,n}]$. We synthesized game level L by placing various game assets, $A = [a_1, a_2, ..., a_n]$, that are characterized by different spatial exploration onto L. The properties of each asset a_i are described in Section 3.3. We evaluated the quality of the generated level L using a total cost function, $C_{\text{Total}}(L)$:

$$C_{\text{Total}}(L) = \mathbf{w}_{\text{Exp}}^T \mathbf{C}_{\text{Exp}} + \mathbf{w}_{\text{Level}}^T \mathbf{C}_{\text{Level}}. \tag{1}$$

where $\mathbf{C}_{\text{Exp}} = [C_{\text{Exp}}^M, C_{\text{Exp}}^V, C_{\text{Exp}}^D]$ is a vector of spatial exploration cost and $\mathbf{w}_{\text{Exp}} = [w_{\text{Exp}}^M, w_{\text{Exp}}^V, w_{\text{Exp}}^D]$ is a vector of weights. C_{Exp}^M, C_{Exp}^V, and C_{Exp}^D encode our spatial exploration considerations. Specifically, C_{Exp}^M denotes the mean spatial exploration of level L, C_{Exp}^V denotes the spatial exploration variance in the level, and with C_{Exp}^D denotes the spatial exploration distribution in level L. $\mathbf{C}_{\text{Level}} = [C_{\text{Level}}^{OA}, C_{\text{Level}}^{AP}, C_{\text{Level}}^{HD}]$ is a vector of level design cost terms, and $\mathbf{w}_{\text{Level}} = [w_{\text{Level}}^{OA}, w_{\text{Level}}^{AP}, w_{\text{Level}}^{HD}]$ is a vector of weights. C_{Level}^{OA}, C_{Level}^{AP}, and C_{Level}^{HD} encode our level design considerations: C_{Level}^{OA} denotes the occupied area ratio between grid cells occupied by asset a_i over the total number of grid cells in level L, C_{Level}^{AP} denotes the adjacent penalty cost that prevents two similar game level assets' placement next to one another, and C_{Level}^{HD} represents the height distribution of asset a_i in the game level L. We provide details for each cost term in the following subsections.

4.1 Spatial Exploration Costs

We used three cost terms to control spatial exploration in a synthesized game level L: *mean spatial exploration*, *spatial exploration variance*, and *spatial exploration distribution*. We describe all spatial exploration costs below.

Mean Spatial Exploration Cost. We synthesized a game level by composing an $|A|$ number of a_i level assets placed on our grid map. We used the mean spatial exploration cost term to compute the difference between a designer-requested mean spatial exploration target value and the mean spatial exploration of level L:

$$C^M_{\text{Exp}}(L) = \left(\frac{1}{|A|}\sum_{a_i}\frac{1}{|\boldsymbol{\mathcal{E}}(a_i)|}\sum_{\varepsilon_{s,t}}\varepsilon_{s,t}(a_i) - \rho_M\right)^2, \quad (2)$$

where $\varepsilon_{s,t}(a_i)$ denotes the spatial exploration value of each cell asset a_i. $|A|$ and $|\boldsymbol{\mathcal{E}}(a_i)|$ return the number of game assets and the cells that each game level asset occupies in L, respectively. Additionally, ρ_M denotes the designer-defined mean target value of spatial exploration. For example, if we assigned a low ρ_M target value, our system would synthesize a game level characterized by low spatial exploration.

Spatial Exploration Variance Cost. We developed the *spatial exploration variance cost* term to compare the spatial exploration variance in game level L with designer-defined target spatial exploration variance. We represented this cost as:

$$C^V_{\text{Exp}}(L) = \left|\frac{1}{|A|}\sum_{a_i}\frac{1}{|\boldsymbol{\mathcal{E}}(a_i)|}\sum_{\varepsilon_{s,t}}\left(\varepsilon_{s,t}(a_i) - \bar{\boldsymbol{\mathcal{E}}}\right)^2 - \rho_V\right|, \quad (3)$$

where $\bar{\boldsymbol{\mathcal{E}}}$ denotes the mean spatial exploration from the generated game level L and ρ_V represents the user-defined target spatial exploration variance that the optimization process should introduce in the synthesized game level L. For example, if we assigned a high ρ_V target value, the synthesized game level would likely comprise assets covering the whole spectrum of spatial exploration assets in our dataset.

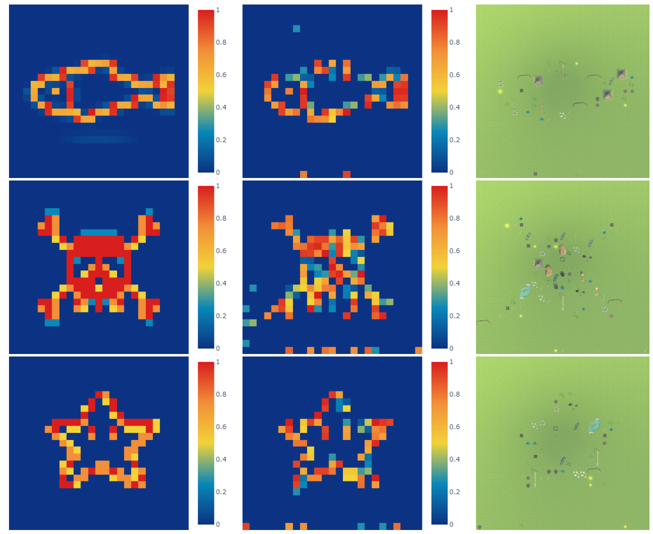
Spatial Exploration Distribution Cost. We developed the *spatial exploration distribution cost* term to compare the spatial exploration distribution across synthesized game level L. Specifically, we provided a level designer with the ability to create spatial exploration distribution pattern D, equal in size to L, which encodes spatial exploration values, such as $\boldsymbol{\mathcal{E}}(D) = [\varepsilon_{1,1}(D), ..., \varepsilon_{m,n}(D)]$, and to use it as an input in order to direct assets' placement on a synthesized level. We provide examples of different spatial exploration distribution patterns and synthesized game levels in Figure 3. This cost computes the difference between the spatial exploration assigned at each cell of synthesized game level L and designer-defined spatial exploration distribution. We applied a Gaussian model to our C^D_{Exp} cost term to penalize deviation and large variation from the desired targets. Specifically, we defined the spatial exploration distribution cost as:

$$C^D_{\text{Exp}}(L) = 1 - \exp\left(-\frac{1}{2\sigma^2}\frac{1}{|L|}\sum_{l_{i,j}}\left(\mathcal{E}(l_{i,j}) - \varepsilon_{i,j}(D)\right)^2\right), \quad (4)$$

where $\mathcal{E}(l_{i,j})$ returns the spatial exploration value of map cell $l_{i,j}$, $\varepsilon_{i,j}(D)$ returns the spatial exploration of input distribution patterns D, and σ controls the spread of the Gaussian penalty function, which we set to $\sigma = 3.00$ through trial testing.

4.2 Level Design Cost

We implemented three cost terms to control level design: *occupied area*, *adjacent penalty*, and *height pattern*. The following subsections explain each of these costs.



(a) (b) (c)

Figure 3: A level designer can provide input into spatial exploration distribution patterns, and our system synthesizes a level by following a spatial exploration distribution input. (a) Input spatial exploration distribution, (b) resulted spatial distribution, and (c) synthesized game level.

Occupied Area Cost. We implemented the *occupied area cost* term to control the ratio between cells $l_{i,j}$ of level L occupied by assets a_i compared to the total number of cells in level L. This cost enables level designers to control how crowded synthesized level L is with assets. We defined the occupied area cost term as:

$$C^{OA}_{\text{Level}}(L) = \left(\frac{1}{|L|}\sum_{l_{i,j}}\Gamma(l_{i,j}) - \rho_{OA}\right)^2, \quad (5)$$

where $\Gamma(l_{i,j})$ returns 1 if asset a_i occupies cell $l_{i,j}$ of a game level; otherwise, it returns 0. $\rho_{OA} \in [0, 1]$ denotes the designer-defined target value. If we set a high target value to ρ_{OA}, asset a_i would occupy most grid cells of the synthesized game level; if we set a low target value to ρ_{OA}, fewer assets would appear in the synthesized game level.

Adjacent Penalty Cost. We implemented the *adjacent penalty cost* term to prevent the system from synthesizing "monotonic" game levels by repeating the same asset next to other instances of that asset. We defined the *adjacent penalty cost* as:

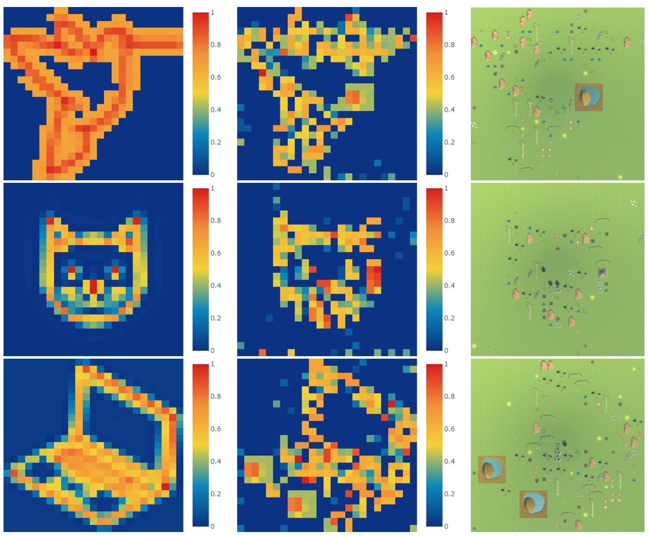
$$C^{AP}_{\text{Level}}(L) = \left(\frac{1}{|A| - 1}\sum_{a_i,a_j}\Pi(a_i, a_j)\right)^2, \quad (6)$$

where (a_i, a_j) represents a pair of level assets and $\Pi(a_i, a_j)$ returns 1 if the two assets are next to each other; otherwise, $\Pi(a_i, a_j)$ returns 0, meaning the two assets are not close to each other.

Height Distribution Cost. We implemented an additional cost term, *height distribution cost*, to allow designers to control the assets' distribution across a synthesized level, this time based on the height of each game asset. For this cost term, the designer inputs 2D heightmap M, which we represent as $\boldsymbol{\mathcal{H}}(M) = [h_{1,1}(M), ..., h_{m,n}(M)]$, and which is equal in size to the grid of level L (i.e., $|M| = |L|$). We then defined the *height distribution cost* term as:

$$C^{HD}_{\text{Level}}(L) = 1 - \exp\left(-\frac{1}{2\sigma^2}\frac{1}{|L|}\sum_{l_{i,j}}\left(\mathcal{H}(l_{i,j}) - h_{i,j}(M)\right)^2\right), \quad (7)$$

where $\mathcal{H}(l_{i,j})$ returns the height value of the asset currently placed in grid cell $l_{i,j}$ and σ controls the spread of the Gaussian penalty function, which we set to $\sigma = 3.00$ through trial testing. We present examples of different height distribution patterns and corresponding synthesized game levels in Figure 4.



(a) (b) (c)

Figure 4: Example synthesized game levels based on different height distribution patterns. (a) Input height distribution, (b) resultant height distribution, and (c) synthesized game level.

4.3 Optimization

Based on target cost function values, our system optimizes the total cost function through a Markov chain Monte Carlo (MCMC) method called "simulated annealing" using a Metropolis-Hastings state-search step. Since our system needs to consider all possible level design outcomes during optimization, we employed the reversible-jump variation of the MCMC technique [10]. To apply simulated annealing, we first defined a Boltzmann-like objective function:

$$f(L) = \exp\left(-\frac{1}{t}C_{\text{Total}}(L)\right), \quad (8)$$

where t encodes the temperature parameter of simulated annealing. During optimization, the system iteratively proposes a new configuration of a level (L') by altering its current configuration (L). The system suggests a new game level (L'), by choosing one of the following moves:

- **Add a level asset:** Our system randomly selects one of the level's assets from our dataset and places it at a randomly chosen location of grid L. If the system places the asset in a position that overlaps with other level assets, the system rejects this proposed design.
- **Remove a level asset:** When the system removes a level asset, it randomly selects one of the level assets currently placed in L and removes it.
- **Replace a level asset:** By replacing a level asset, our system randomly selects one of the level assets in current level configuration L and replaces it with a randomly selected level asset from our dataset. To prevent overlapping with an asset in L, the system only selects an asset from the dataset that is equal in size to the asset selected from L.

We set the probability of adding a level asset to $p_{\text{add}} = .40$, the probability of removing a level asset to $p_{\text{remove}} = .20$, and the probability of replacing a level asset to $p_{\text{replace}} = .40$. Through these probabilities, our system chooses to add a level asset and replace a level asset more often than choosing to remove an asset.

By applying one of these moves, our system proposes a game level (L') and compares the total cost of the proposed game level (L') with the total cost of the current game level (L) to determine whether it accepts the proposed game level (L') or keeps the current game level (L). To ensure balanced, trans-dimensional optimization, we defined the probability of each move. Our system computes the probability of adding a level asset as:

$$p_{\text{add}}(L'|L) = \min\left(1, \frac{p_{\text{remove}}}{p_{\text{add}}}\frac{U - |L|}{|L'|}\frac{f(L')}{f(L)}\right). \quad (9)$$

It computes the probability of removing a level asset as:

$$p_{\text{remove}}(L'|L) = \min\left(1, \frac{p_{\text{add}}}{p_{\text{remove}}}\frac{|L|}{U - |L'|}\frac{f(L')}{f(L)}\right). \quad (10)$$

It computes the probability of replacing a level asset move as:

$$p_{\text{replace}}(L'|L) = \min\left(1, \frac{f(L')}{f(L)}\right). \quad (11)$$

For the above formulations, we set an upper limit on the amount frequency at which each level asset could be chosen during optimization using the variable U. We assumed that our system could select each asset up to U_i times, rather than an infinite number of times. Thus, our system synthesizes a level of up to $U = \sum_i U_i$ game level objects. In our implementation, we used $U = 300$ for all level assets.

We applied simulated annealing to explore our solution space effectively. Simulated annealing allowed us to use a temperature parameter (t) to control the acceptance probability of the synthesized game level configurations. If the temperature parameter were high, the system would aggressively explore the whole solution space. If the temperature parameter were low, the optimizer would become greedier. We initialized the temperature parameter as $t = 1.00$ at the beginning of optimization. In each iteration, we multiplied the temperature parameter by $t^* = .99975$, and the optimization process terminated when the change in $C_{\text{Total}}(L)$ was less than .25% over the last 100 iterations after 5,000 iterations.

Unless we specified otherwise, we set the weight of the mean spatial exploitation cost to $w^M_{\text{Exp}} = 1.00$, the weight of the spatial exploration variance cost to $w^V_{\text{Exp}} = .80$, and the weight of the spatial exploration distribution cost to $w_{\text{Exp}}^{D} = .10$. Moreover, for the prior cost terms, we set the weight of the occupied area cost to $w_{\text{Level}}^{OA} = .40$, the weight of the adjacent penalty cost to $w_{\text{Level}}^{AP} = .20$, and the weight of the height distribution cost to $w_{\text{Level}}^{HD} = .40$. Obviously, our design decisions prioritized the mean spatial exploitation cost and the spatial exploration variance cost over all other cost terms. However, a level designer can easily adjust the cost terms' heights to explore various other design outputs.

5 EVALUATING SPATIAL EXPLORATION

We conducted a user study to evaluate whether our proposed method to synthesize game levels would trigger a designer-specified amount of spatial exploration among study participants. In the following subsections, we describe our methodology for this study.

5.1 Participants

For this between-group study, we conducted an a priori power analysis [4] to determine a suitable sample size using the G*Power [7] software version 3.10. These calculations were based on a small effect size of .20, a .90 power, an $\alpha = .05$, and three groups. This analysis resulted in a recommended sample size of 216 participants.

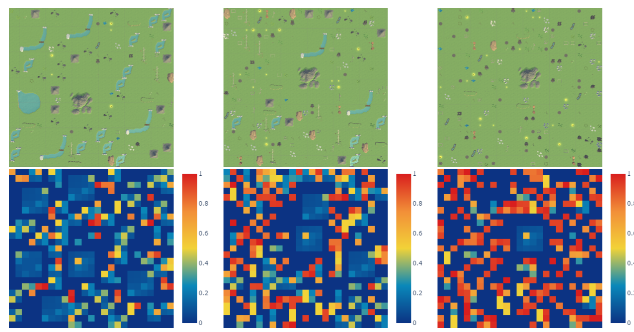
To recruit participants, we emailed undergraduate students in our department. In total, 225 students (75 per experimental condition) volunteered to participate (age: $M = 21.36$, $SD = 2.28$). Of this sample, 180 participants were male, 39 were female, and six preferred not to disclose their gender. Moreover, 33 participants identified as novice game players, 96 as casual players, 49 as core players, and 47 as expert players. Finally, when participants stated how often they had played video games in the last year, two indicated never, six indicated less than one hour in total, eight indicated one hour, 38 indicated one hour monthly on average, 69 indicated one hour weekly, 54 indicated one hour daily, and 54 indicated more than one hour daily.

5.2 Experimental Conditions

We developed three game levels to serve as experimental conditions in order to determine whether our method could synthesize game levels that influence participants' spatial exploration. As we mentioned earlier, we used a between-group study design; thus, different participant groups played different levels, which we developed to represent the following three conditions:

- **Low Spatial Exploration (LSE):** For the *low spatial exploration* condition, we set a target for mean spatial exploration at $\rho_M = .25$. Under this condition, we expected participants to wander around the level more than they engaged with the game level assets.
- **Medium Spatial Exploration (MSE):** For the *medium spatial exploration* condition, we set a target for mean spatial exploration at $\rho_M = .50$. Under this condition, we expected participants to spend roughly equal time wandering around the level and engaging with the game level assets.
- **High Spatial Exploration (HSE):** For the *high spatial exploration* condition, we set a target for mean spatial exploration at $\rho_M = .75$. Under this condition, we expected participants to spend more time exploring the game level assets than wandering around the level.

For all three conditions, we set a spatial exploration variance target of $\rho_V = .10$ and an occupied area target of $\rho_{OA} = .50$. We assigned a weight of $w_{\text{Exp}}^{D} = .00$ to the spatial exploration distribution cost and a weight of $w_{\text{Exp}}^{HD} = .00$ for the height distribution cost; therefore, our system does not consider these terms during optimization. We present the three levels we used in this study in Figure 5. Finally, we used a 25×25 grid as the map for our open-world game (as mentioned, each grid cell was 25×25 Unity units in size). We realized during our pilot study that this grid size constituted a sufficient size to ensure that a level was not too small and allow the player enough space to move around and explore. All game levels and our implementations can be found on our project's website and downloaded from there.



(a) (b) (c)

Figure 5: A top-down view of the three different game levels used in our user study. From left to right: (a) low spatial exploration, (b) medium spatial exploration, and (c) high spatial exploration. The heatmapping represents how spatial exploration was distributed across the open-world levels.

5.3 Measurements

We collected data on participants' spatial exploration behavior. Specifically, *spatial exploration* refers to the time each participant spent inside each cell of each game level asset a_i over their total time spent in a level. To collect these data, we used the position of the avatar controlled by participants and computed its total time spent inside each grid cell. We expected that, the more time a player spent inside a cell, the more curious they were to explore the related asset.

We also collected self-reported ratings through a post-game survey that we distributed to our participants. This survey collected demographic information and selected measurements from the *Game User Experience Satisfaction Survey* (GUESS). GUESS was validated by Phan et al. [34], and it examines game experience across several areas. Its measurements can be used independently, depending on a project's needs. For our study, we included GUESS's *enjoyment*, *creative freedom*, *play engrossment*, and *personal gratification* measurements to evaluate how participants perceived and experienced our synthesized game levels.

5.4 Procedure

We conducted this study remotely to minimize the spread of coronavirus disease 2019 (COVID-19). Specifically, we emailed participants instructions on how to download our game and complete the study. We provided participants with written, step-by-step instructions (playing the game, locating a saved `.csv` file, uploading their save file to our survey website after finishing the game, and completing our online survey in Qualtrics). Moreover, we asked participants to pay attention to our in-game instructions, which directed them step-by-step on how to complete the study. Additionally, we provided in-game instructions to participants regarding the game's controls, which have been shown that improve performance, players' experience, and players' intrinsic motivation [17]. After downloading the game and before completing the study, we asked participants to consent to participation by signing an online form. Once enough participants had indicated their interest in volunteering for our study, we randomly assigned them to one of our experimental conditions.

During participants' gameplay, our game application recorded their spatial exploration and saved it to a `.csv` file. Participants played our game for 10 minutes. After 10 minutes, a message was displayed onscreen, instructing them to proceed with the study. We instructed participants to press the **X** key on their keyboard; then, a browser window opened and loaded our survey website. If this browser window did not load, our initial instructions provided participants with all the information they needed to access the survey website. Participants were asked to answer all questions on the survey page. Moreover, we provided a text box area for participants to provide comments they thought would be helpful for future improvement of our game and gameplay experience. Finally, the survey website instructed participants to use an "upload" button to upload the `.csv` file that had been generated during their gameplay immediately after they had pressed the **X** key. After participants had submitted their survey responses, they were free to play our game as long as they wished.

5.5 Results

We used a one-way analysis of variance (ANOVA) to explore potential differences across the examined conditions. Both Shapiro-Wilk tests at the 5% level and a graphs of the residuals using Q-Q plots indicated our data's normality. We used a p-value of $< .05$ to denote statistical significance. Finally, we used Bonferroni-corrected estimates for our post hoc comparisons.

5.5.1 Spatial Exploration. Analysis of participants' spatial exploration measurements indicated significant differences across the examined conditions ($F[2, 222] = 23.706$, $p = .0001$). The results of our post hoc comparison showed that spatial exploration under the LSE condition ($M = .68$, $SD = .66$) was statistically significantly lower than under the MSE condition ($M = .72$, $SD = .77$) at $p = .001$ and the HSE condition ($M = .76$, $SD = .64$) at $p = .001$. Moreover, spatial exploration under the MSE condition was statistically significantly lower than under the HSE condition at $p = .043$.

In addition to our statistical analysis of spatial exploration, we also sought to understand how participants explored had visually explored our synthesized game levels. Accordingly, we averaged participants' spatial exploration data for each cell of the game level grid. Later, we used spatial exploration heatmaps of our synthesized levels and of participants' data and computed their differences, which we present in Figure 6.

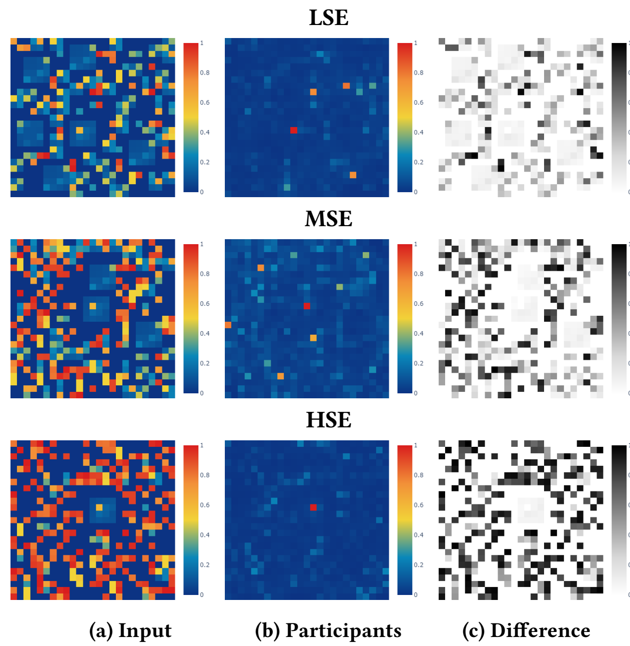


Figure 6: Heatmaps of spatial exploration in our synthesized game levels (LSE, MSE, and HSE). Participants' spatial exploration data are plotted on a heatmap, along with the differences between the spatial exploration of the synthesized game levels and participants' data.

5.5.2 GUESS Survey. We also analyzed participants' self-reported ratings based on the GUESS survey. Unfortunately, we could not identify statistically significant results for all four examined measurements (*enjoyment*, *creative freedom*, *play engrossment*, and *personal gratification*). Specifically, ANOVAs revealed: $F(2, 222) = .149$, $p = .861$ for *enjoyment*; $F(2, 222) = .191$, $p = .826$ for *creative freedom*; $F(2, 222) = .005$, $p = .995$ for *play engrossment*; and $F(2, 222) = 1.091$, $p = .338$ for *personal gratification*.

5.6 Discussion

Our user study sought to explain whether our method could synthesize game levels that trigger designer-specified spatial exploration behavior among participants. Accordingly, we collected data on participants' in-game spatial exploration and their self-reported ratings through a post-game survey.

Based on *spatial exploration* measurements, we found that our method can synthesize game levels that trigger different spatial exploration behaviors among participants for all three of our experimental conditions (LSE, MSE, and HSE). Initially, our results suggested that our method could synthesize game levels that impact the spatial exploration behavior of participants. However, although we obtained statistically significant results, we observed a large offset between the requested mean spatial exploration targets

(ρ_M = .25 for LSE, ρ_M = .50 for MSE, and ρ_M = .75 for HSE) and participants' mean values (M = .68 for LSE, M = .72 for MSE, and M = .76 for HSE)—especially under the LSE and MSE conditions. Specifically, the results we obtained for the HSE condition are close to the targets we assigned to ρ_M (offset = .01). However, our LSE and MSE results strongly suggest that the offset between the mean spatial exploration target ρ_M and participants' actual spatial exploration data increased dramatically (offset = .22 for MSE and offset = .43 for LSE).

However, we understand from previously published studies that such offsets are present in experience-driven designs [24, 30, 32, 33] between target values and participants' actual performance or perception. We interpreted the large offset values obtained in our study through our game's main instruction provided to participants. Specifically, we instructed participants to *"feel free to explore this game level;"* thus, we intentionally instructed participants to explore. Likely, they tried to explore as much of the level as possible instead of exploring the game assets. We also observed this in the difference between the input and resultant heatmaps in Figure 6. Instead of observing more bright areas in the computed difference, we saw more dark cells, indicating that participants did not try to focus on assets but, rather, focused on exploring the level. However, across all three conditions, one observation was remarkable: near the center of each level, the optimizer placed a tall mountain asset, and participants' data clearly shows a red cell near the center of all three levels, indicating that almost all participants visited this mountain (EXP category) and spent more time around this asset. This finding is similar to what Gómez-Maureira [9] reported in their study, which we used as a basis for our paper. However, our study expands on Gómez-Maureira's [9] findings about how participants explored different asset types (EXP, OBS, OOP, and SPC). Specifically, our statistical analysis revealed that participants' spatial exploration could be influenced by asset type. We found that assets of the EXP type were explored more, while assets of the OBS type were explored the least and assets of the OOP and SPC were explored to middling extents between EXP and OBS. This finding should be considered by game level designers who wish to induce players' spatial exploration.

In addition to tracking in-game spatial exploration, we also asked participants to respond to a post-game survey that collected different variables regarding their gameplay experience. Our statistical analysis did not reveal significant differences for all four examined GUESS measurements (*enjoyment*, *creative freedom*, *play engrossment*, and *personal gratification*). We expected this outcome because we specifically designed three open-world levels that share the same assets and, as we mentioned, instructed participants to explore the levels without providing a specific target (e.g., collecting coins or observing their preferred assets). Because the only difference observed across the three levels was the assets selected and placed by our system, this difference insufficiently reflected differences in participants' gameplay experiences. However, our lack of significant results regarding participants' gameplay experience does not invalidate our method. Our main objective was to induce different spatial exploration behaviors among participants, rather than synthesizing an enjoyable experience for participants.

Additionally, we found some similarities across the three experimental conditions, based on selected comments from participants. Regarding game objectives, participants wrote comments such as: *"Great game but consider adding objectives or goals that the character can accomplish."*, *"I was not sure what the goal of the game was."*, and *"While creative and fun for a while. The game simply had no objective. At some point you will become bored of exploring because you will have explored everything. There needs to be objectives in order to keep the player interested."* Regarding interaction in the game, participants wrote: *"I think the game needs some interactive elements to make it more interesting."* and *"There was no objective, no interact button, no lore, there was nothing in the game."* Finally, regarding the exploration part of our game, participant comments included: *"It was interesting to explore the map, but since my game had no objective or progression, it became very boring after exploring the whole map."* and *"There was not any progression. I was just exploring some assets and that was pretty much it for me."*

We found all participant comments reasonable and informative. Our game did not provide any goal to participants. In contrast, participants were left to merely explore an open-world level. From an experimental perspective, we think we made the right decisions since we wanted participants to focus on level assets and explore the levels based on their curiosity, instead of providing an objective that could have distracted them from their curiosity in exploring the map. Given our findings and participants' comments, researchers should conduct additional studies to further examine how goals or targets can influence players' spatial exploration in games.

Alongside the limitations of our project revealed by participants' comments, we noted a few more limitations, which we hope will benefit future researchers interested in studying spatial exploration in games. One such limitation is that the current total cost function considered only assets' placement, instead of the aesthetic compatibility of nearby assets. Including an additional cost term in this regard would help our method improve synthesized level aesthetics. Additionally, a dataset comprising more than 32 level assets should be developed and annotated. Although we designed open-world levels, assets were evidently repeated in these synthesized levels, reducing curiosity-driven exploration behavior. Therefore, introducing more variations to synthesized game levels increase players' enjoyment. Additionally, in this project, we demonstrated a simple approach to synthesizing game levels that we characterized as highly structured, based on a 2D map. Critically, additional level designs—such as mazes and dungeons—must be explored to explain whether and how such layouts can influence players' spatial exploration behavior. Finally, we developed a game with no actual objective, as we mentioned before. Therefore, incorporating some goals, similar to other games—such as collecting coins—would provide additional insights into players' curiosity when exploring a level. Finally, we only included a single asset in each level during our annotation phase. We think an "antagonistic" annotation process—in which multiple assets are placed in a level—could provide more reliable annotation data, especially when a level is designed using multiple assets. Understanding players' curiosity about exploring a level asset against other assets of that game level could help refine our procedural level design method to encourage spatial exploration.

6 CONCLUSIONS AND FUTURE WORK

In this project, we aimed to procedurally generate game levels that induce players' designer-defined spatial exploration. To the best of our knowledge, we have developed the first method that attempts to automatically synthesize game levels by considering players' spatial exploration. Although we achieved the project's primary goal of validating our method's ability to automatically synthesize game levels that induce different spatial exploration behavior through a user study, we also received several comments from participants that revealed project limitations. In a future work, we plan to address this feedback and the mentioned limitations discussed in the previous section by revising our method. We also plan to further explore the potential of generalizing such a method across game genres. Procedural level design targeting designer-specified spatial exploration behavior has significant potential and applicability in games. Therefore, the research community should consider it when developing experience-driven procedural game levels.

REFERENCES

[1] Cameron Browne and Frederic Maire. 2010. Evolutionary game design. *IEEE Transactions on Computational Intelligence and AI in Games* 2, 1 (2010), 1–16.

[2] Luigi Cardamone, Daniele Loiacono, and Pier Luca Lanzi. 2011. Interactive evolution for the procedural generation of tracks in a high-end racing game. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*. 395–402.

[3] Luigi Cardamone, Georgios N Yannakakis, Julian Togelius, and Pier Luca Lanzi. 2011. Evolving interesting maps for a first person shooter. In *European Conference on the Applications of Evolutionary Computation*. Springer, 63–72.

[4] Jacob Cohen. 2013. *Statistical power analysis for the behavioral sciences*. Routledge.

[5] Kate Compton and Michael Mateas. 2006. Procedural level design for platform games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 2. 109–111.

[6] Greg Costikyan. 2013. *Uncertainty in games*. Mit Press.

[7] Franz Faul, Edgar Erdfelder, Albert-Georg Lang, and Axel Buchner. 2007. G* Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences. *Behavior research methods* 39, 2 (2007), 175–191.

[8] Marcello A Gómez-Maureira and Isabelle Kniestedt. 2019. Exploring video games that invoke curiosity. *Entertainment Computing* 32 (2019), 100320.

[9] Marcello A Gómez-Maureira, Isabelle Kniestedt, Max Van Duijn, Carolien Rieffe, and Aske Plaat. 2021. Level Design Patterns That Invoke Curiosity-Driven Exploration: An Empirical Study Across Multiple Conditions. *Proceedings of the ACM on Human-Computer Interaction* 5, CHIPLAY (2021), 1–32.

[10] Peter J Green. 1995. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* 82, 4 (1995), 711–732.

[11] John Harris. 2007. Game design essentials: 20 open world games. *Gamasutra–The Art and Business of Making Games* 26 (2007).

[12] Erin Jonathan Hastings, Ratan K Guha, and Kenneth O Stanley. 2009. Automatic content generation in the galactic arms race video game. *IEEE Transactions on Computational Intelligence and AI in Games* 1, 4 (2009), 245–263.

[13] Mark Hendrikx, Sebastiaan Meijer, Joeri Van Der Velden, and Alexandru Iosup. 2013. Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 9, 1 (2013), 1–22.

[14] Danial Hooshyar, Moslem Yousefi, Minhong Wang, and Heuiseok Lim. 2018. A data-driven procedural-content-generation approach for educational games. *Journal of Computer Assisted Learning* 34, 6 (2018), 731–739.

[15] Kenneth Hullett and Jim Whitehead. 2010. Design patterns in FPS levels. In *proceedings of the Fifth International Conference on the Foundations of Digital Games*. 78–85.

[16] Martin Jennings-Teats, Gillian Smith, and Noah Wardrip-Fruin. 2010. Polymorph: A model for dynamic level generation. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 6. 138–143.

[17] Dominic Kao, Alejandra J Magana, and Christos Mousas. 2021. Evaluating Tutorial-Based Instructions for Controllers in Virtual Reality Games. *Proceedings of the ACM on Human-Computer Interaction* 5, CHI PLAY (2021), 1–28.

[18] Daniel Karavolos, Antonios Liapis, and Georgios Yannakakis. 2019. A Multifaceted Surrogate Model for Search-Based Procedural Content Generation. *IEEE Transactions on Games* 13, 1 (2019), 11–22.

[19] Todd B Kashdan, Melissa C Stiksma, David J Disabato, Patrick E McKnight, John Bekier, Joel Kaji, and Rachel Lazarus. 2018. The five-dimensional curiosity scale: Capturing the bandwidth of curiosity and identifying four unique subgroups of curious people. *Journal of Research in Personality* 73 (2018), 130–149.

[20] Manuel Kerssemakers, Jeppe Tuxen, Julian Togelius, and Georgios N Yannakakis. 2012. A procedural procedural level generator generator. In *2012 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 335–341.

[21] Ahmed Khalifa, Fernando de Mesentier Silva, and Julian Togelius. 2019. Level design patterns in 2D games. In *2019 IEEE Conference on Games (CoG)*. IEEE, 1–8.

[22] Christoph Klimmt. 2003. Dimensions and determinants of the enjoyment of playing digital games: A three-level model. In *Level up: Digital games research conference*. 246–257.

[23] Schulamith Kreitler, Edward Zigler, and Hans Kreitler. 1975. The nature of curiosity in children. *Journal of school psychology* 13, 3 (1975), 185–200.

[24] Wanwan Li, Biao Xie, Yongqi Zhang, Walter Meiss, Haikun Huang, and Lap-Fai Yu. 2020. Exertion-aware path generation. *ACM Trans. Graph.* 39, 4 (2020), 115.

[25] Antonios Liapis, Georgios Yannakakis, and Julian Togelius. 2013. Designer modeling for personalized game content creation tools. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 9. 11–16.

[26] Antonios Liapis, Georgios Yannakakis, and Julian Togelius. 2013. Towards a generic method of evaluating game levels. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 9. 30–36.

[27] Antonios Liapis, Georgios N Yannakakis, and Julian Togelius. 2012. Adapting models of visual aesthetics for personalized content creation. *IEEE Transactions on Computational Intelligence and AI in Games* 4, 3 (2012), 213–228.

[28] Jordan A Litman. 2008. Interest and deprivation factors of epistemic curiosity. *Personality and individual differences* 44, 7 (2008), 1585–1595.

[29] Jordan A Litman. 2010. Relationships between measures of I-and D-type curiosity, ambiguity tolerance, and need for closure: An initial test of the wanting-liking model of information-seeking. *Personality and Individual Differences* 48, 4 (2010), 397–402.

[30] Huimin Liu, Minsoo Choi, Dominic Kao, and Christos Mousas. 2022. Synthesizing Game Levels for Collaborative Gameplay in a Shared Virtual Environment. *ACM Transactions on Interactive Intelligent Systems (TiiS)* (2022).

[31] Huimin Liu, Zhiquan Wang, Angshuman Mazumdar, and Christos Mousas. 2021. Virtual reality game level layout design for real environment constraints. *Graphics and Visual Computing* 4 (2021), 200020.

[32] Huimin Liu, Zhiquan Wang, Christos Mousas, and Dominic Kao. 2020. Virtual reality racket sports: Virtual drills for exercise and training. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 566–576.

[33] Christos Mousas, Claudia Krogmeier, and Zhiquan Wang. 2021. Photo Sequences of Varying Emotion: Optimization with a Valence-Arousal Annotated Dataset. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 11, 2 (2021), 1–19.

[34] Mikki H Phan, Joseph R Keebler, and Barbara S Chaparro. 2016. The development and validation of the game user experience satisfaction scale (GUESS). *Human factors* 58, 8 (2016), 1217–1247.

[35] David Plans and Davide Morelli. 2012. Experience-driven procedural music generation for games. *IEEE Transactions on Computational Intelligence and AI in Games* 4, 3 (2012), 192–198.

[36] Mike Preuss, Antonios Liapis, and Julian Togelius. 2014. Searching for good and diverse game levels. In *2014 IEEE Conference on Computational Intelligence and Games*. IEEE, 1–8.

[37] Jesse Schell. 2008. *The Art of Game Design: A book of lenses*. CRC press.

[38] Noor Shaker, Julian Togelius, and Mark J Nelson. 2016. *Procedural content generation in games*. Springer.

[39] Peizhi Shi and Ke Chen. 2017. Learning constructive primitives for real-time dynamic difficulty adjustment in Super Mario Bros. *IEEE Transactions on Games* 10, 2 (2017), 155–169.

[40] Tianye Shu, Jialin Liu, and Georgios N Yannakakis. 2021. Experience-driven PCG via reinforcement learning: A Super Mario Bros study. In *2021 IEEE Conference on Games (CoG)*. IEEE, 1–9.

[41] Gillian Smith, Ryan Anderson, Brian Kopleck, Zach Lindblad, Lauren Scott, Adam Wardell, Jim Whitehead, and Michael Mateas. 2011. Situating quests: Design patterns for quest and level design in role-playing games. In *International Conference on Interactive Digital Storytelling*. Springer, 326–329.

[42] Nathan Sorenson, Philippe Pasquier, and Steve DiPaola. 2011. A generic approach to challenge modeling for the procedural creation of video game levels. *IEEE Transactions on Computational Intelligence and AI in Games* 3, 3 (2011), 229–244.

[43] Penelope Sweetser. 2008. Emergence in games. *Game Connect Asia Pacific 2008* (2008).

[44] Penelope Sweetser and Janet Wiles. 2005. Scripting versus emergence: issues for game developers and players in game environment design. *International Journal of Intelligent Games and Simulation* 4, 1 (2005), 1–9.

[45] Alexandra To, Safinah Ali, Geoff F Kaufman, and Jessica Hammer. 2016. Integrating Curiosity and Uncertainty in Game Design. *Digra/fdg* (2016).

[46] Julian Togelius and Jurgen Schmidhuber. 2008. An experiment in automatic game design. In *2008 IEEE Symposium On Computational Intelligence and Games*. Citeseer, 111–118.

[47] Julian Togelius, Jim Whitehead, and Rafael Bidarra. 2011. Guest editorial: Procedural content generation in games. *IEEE Transactions on Computational Intelligence and AI in Games* 3, 03 (2011), 169–171.
[48] Julian Togelius, Georgios N Yannakakis, Kenneth O Stanley, and Cameron Browne. 2011. Search-based procedural content generation: A taxonomy and survey. *IEEE Transactions on Computational Intelligence and AI in Games* 3, 3 (2011), 172–186.
[49] Christopher W Totten. 2019. *Architectural Approach to Level Design*. CRC Press.
[50] Evan James Williams. 1949. Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Chemistry* 2, 2 (1949), 149–168.
[51] Biao Xie, Yongqi Zhang, Haikun Huang, Elisa Ogawa, Tongjian You, and Lap-Fai Yu. 2018. Exercise intensity-driven level design. *IEEE transactions on visualization and computer graphics* 24, 4 (2018), 1661–1670.
[52] Georgios N Yannakakis and Ana Paiva. 2014. Emotion in games. *Handbook on affective computing* 2014 (2014), 459–471.
[53] Georgios N Yannakakis and Julian Togelius. 2011. Experience-driven procedural content generation. *IEEE Transactions on Affective Computing* 2, 3 (2011), 147–161.