Synthesizing Game Levels for Collaborative Gameplay in a Shared Virtual Environment

HUIMIN LIU, Purdue University, U.S.A.
MINSOO CHOI, Purdue University, U.S.A.
DOMINIC KAO, Purdue University, U.S.A.
CHRISTOS MOUSAS, Purdue University, U.S.A.

Fig. 1. Our method synthesizes a game level in which participants collaborate in a shared virtual environment to play a game.

We developed a method to synthesize game levels that accounts for the degree of collaboration required by two players to finish a given game level. We first asked a game level designer to create playable game level chunks. Then, two artificial intelligence (AI) virtual agents driven by behavior trees played each game level chunk. We recorded the degree of collaboration required to accomplish each game level chunk by the AI virtual agents and used it to characterize each game level chunk. To synthesize a game level, we assigned to the total cost function cost terms that encode both the degree of collaboration and game level design decisions. Then, we used a Markov-chain Monte Carlo optimization method, called simulated annealing, to solve the total cost function and proposed a design for a game level. We synthesized three game levels (low, medium, and high degrees of collaboration game levels) to evaluate our implementation. We then recruited groups of participants to

Authors' addresses: Huimin Liu, Purdue University, 401 N. Grant St., West Lafayette, Indiana 47907, U.S.A., liu2833@purdue.edu; Minsoo Choi, Purdue University, 401 N. Grant St., West Lafayette, Indiana 47907, U.S.A., choi714@purdue.edu; Dominic Kao, Purdue University, 401 N. Grant St., West Lafayette, Indiana 47907, U.S.A., kaod@purdue.edu; Christos Mousas, Purdue University, 401 N. Grant St., West Lafayette, Indiana 47907, U.S.A., cmousas@purdue.edu.

2160-6455/2022/8-ART
https://doi.org/10.1145/3558773

play the game levels to explore whether they would experience a certain degree of collaboration and validate whether the AI virtual agents provided sufficient data that described the collaborative behavior of players in each game level chunk. By collecting both in-game objective measurements and self-reported subjective ratings, we found that the three game levels indeed impacted the collaboration gameplay behavior of our participants. Moreover, by analyzing our collected data, we found moderate and strong correlations between the participants and the AI virtual agents. These results show that game developers can consider AI virtual agents as an alternative method for evaluating the degree of collaboration required to finish a game level.

CCS Concepts: • **Human-centered computing** → **Collaborative interaction**; • **Applied computing** → **Computer games**; • **Computing methodologies** → *Intelligent agents.*

Additional Key Words and Phrases: game level, chunks, collaboration, AI agents, behavior trees, optimization

1 INTRODUCTION

In our daily lives, we collaborate with others on various tasks in various ways. According to Webster's Dictionary, "*collaboration*"[1] refers to "**the work and activity of a number of persons who individually contribute toward the efficiency of the whole.**" In addition to real-world collaborative tasks that people perform in their everyday lives (e.g., two people collaborate to rearrange a couch), people also perform tasks in virtual worlds and video games (e.g., two people collaborate to catch an enemy). Although collaborative experiences in humans' daily lives are relatively common, the evolutionary foundations of humans' collaborative skills remain unclear [44].

In games and VR applications, the tasks requiring users to collaborate and the degree of collaboration required to accomplish a given task are manually built or programmed by the game's designers. However, a game designer can design hundreds of game levels that share similar game level chunks. For example, a game level designer can synthesize platform games (e.g., games similar to *Super Mario Land*[2]) by repeating various predesigned game level chunks. In addition, the designer is responsible for fine-tuning the degree of collaboration required for each game level, which is a tedious and time-consuming process. To overcome these issues, we propose a pipeline that automatically characterizes the degree of collaboration of game level chunks and synthesizes game levels with designer-defined degrees of collaboration targets (Fig. 1). As a result, a game level designer can request game levels with different degrees of collaboration. The designer can later edit the synthesized game level if needed, automating the whole process and minimizing the time required to design the game levels.

In this project, we targeted the "shared goal" [1, 70] and "mutual benefit" [65] aspects of collaboration. In particular, we thought that providing a shared goal to the players (finishing the game level) would work as a force that holds players together and allows them to coordinate their efforts and work together toward a mutual benefit. According to Uhlaner et al. [72], when there are strong shared goals, players are more likely to prioritize group needs over personal needs. In addition, there tends to be more cooperation and collaboration when there are strong shared goals, and players are more likely to defer personal benefits for collective benefits. Shared goals focus and coordinate strategic action toward the mutual benefit, increasing the likelihood that players can simultaneously fulfill individual and group goals.

The proposed method is divided into three parts. First, a game level designer is responsible for designing playable game level chunks. Second, artificial intelligence (AI) virtual agents are implemented to play the game level chunks. We collect data from these agents and use them to characterize the degree of collaboration of each game level chunk. Third, by developing cost terms that encode various design decisions, our method automatically synthesizes a game level that fulfills all designer-specified design decisions. Such a formulation allows our system to synthesize several variations of game levels that satisfy the designer-defined parameters in a few seconds,

[1] https://www.merriam-webster.com/thesaurus/collaboration
[2] https://www.mariowiki.com/Super_Mario_Land

offering variability across game levels. According to the literature [40, 41, 80], such variability is important for keeping players engaged during gameplay.

The scope of this project was twofold. First, we aimed to validate whether the proposed method automatically synthesized game levels with different degrees of collaboration assigned to them and understand how players changed their gameplay behavior and perceived these different degrees of collaboration in the game levels. Second, we aimed to explore whether AI virtual agents can be used to characterize the collaborative behavior of game level chunks and, thereby, provide sufficient data that describes the collaborative behavior of players in each game level chunk. To accomplish these aims, we conducted a user study to collect data from participants. For our user study, we requested that our optimizer synthesize game levels requiring low, medium, and high degree of collaboration. We collected various in-game measurements during the gameplay. Moreover, we asked the participants to respond according to the scale we developed for this project. The obtained results indicated that our method could synthesize the game levels in which the participants collaborated differently across the three examined conditions (low, medium, and high degrees of collaboration). In addition, we evaluated the ability of the AI virtual agents to provide data that reflected the degree of collaboration required by the participants. The analysis results showed that the participants followed a parallel collaboration pattern with the AI virtual agents, indicating that game designers can use such agents as an alternative method for evaluating the degree of collaboration needed to complete a given game level. In addition to the positive findings of our study, we also discuss some limitations to guide future research in automatic game level design for collaborative gameplay.

The rest of the paper is organized as follows. In Section 2, we present related work on collaborative games and virtual reality experiences. In Section 3, we describe the preliminary remarks of our project. In Section 4, we explain the formulation of the game level synthesis and the optimization process. In Section 5, we outline the conducted user study and discuss our findings. In Section 6, we review the limitations of our method. Finally, in Section 7, we present our conclusions and potential future research directions.

2 RELATED WORK

Computer games encode problem-solving activities in which players build a strategy to overcome the difficulties they face [57], drawing on prior problem-solving knowledge as they explore the solution space for a given problem [33]. According to Sedano et al. [58], collaborative games encode activities in which the players must work together toward a common outcome. This means that the players should work collectively to identify the dominant strategy for a given in-game problem. Most multiplayer games incorporate both collaborative and competitive mechanics. Examples of games that require collaboration between players are *Portal 2*,[3] *Trine*,[4] and *Keep Talking and Nobody Explodes*.[5] In *Keep Talking and Nobody Explodes*, the players need to diffuse a bomb. One player is responsible for explaining how to defuse the bomb by using the provided manual, and the other player is responsible for performing the necessary operation. Providing the option for two or more players to collaborate toward achieving a common goal defines the subgenre of collaborative gameplay.

One of the immensely popular and largest emerging multiplayer game genres that also encode collaboration is the Multiplayer Online Battle Arena (MOBA) [47], e.g., the *League of Legends*[6] game. In such games, two teams of players compete to destroy each other's base. The individual players act collectively, while the teams coordinate to meet shared goals [71]. Additionally, Massively Multiplayer Online Role-Playing Games (MMORPGs), such as the *World of Warcraft*,[7] allow many players to collaborate in various tasks, such as fighting a dragon. According

[3]https://www.thinkwithportals.com/
[4]https://www.frozenbyte.com/games/
[5]https://keeptalkinggame.com/
[6]https://en.wikipedia.org/wiki/League_of_Legends
[7]https://en.wikipedia.org/wiki/World_of_Warcraft

to Wikipedia's list of cooperative video games,[8] some MMORPGs can be played by players ranging from two, such as *Space Duel*[9] and *Sky Force*,[10] to 128, such as *Freelancer*[11] and *The Forest*.[12]

Zagal et al. [81] explored how players who work together influence a game's design by analyzing collaborative board games. They found that some tension between collaboration and selfish play is required to create an interesting collaborative game even though the players ultimately share the same goal and always win or lose as a group. This tension can facilitate discussions about how to reach the shared goal. Zea et al. [82] explored how game level designers can use collaborative learning requirements as game design guidelines. They proposed guidelines to help developers create more efficient collaborative games, such as "give players a common goal and shared rewards," "require a minimal score of each player before the group can progress, but also give the players enough information to enable helping," "make players accountable for their actions, for example by showing their individual results to the group," "guide group members towards social interactions, for example require consensus to foster discussions," and "establish a rotating leader role."

Rocha et al. [53] proposed various methods to force collaboration among the game players. Among them, we can distinguish between the "shared goals" method, in which cooperating players have similar (or identical) objectives that they must complete, putting them on the same pathway toward their goals, and the "complementary" and "synergies between abilities" methods, both of which involve asymmetry between the two (or more) players and their abilities. Seif El-Nasr et al. [59] found additional patterns that define collaboration in commercial games. Specifically, by analyzing 14 games, they found patterns such as "players interacting with the same object," "shared puzzles or characters," "enemies specifically targeting separated players," "automatic vocalization," and "limited (shared) resources." Moreover, through an evaluation process, they validated the importance of such patterns in forming collaborative gameplay. In a similar vein, Reuter et al. [3] introduced game design patterns for collaborative player interactions. They analyzed 15 well-known games from different genres and extracted the patterns used to guide collaborative game designs to foster interaction between players. Later, they classified the interactions into several dimensions (e.g., spatial and temporal). Lastly, to address the issue of authoring collaborative multiplayer games, Reuter [51] conceptualized an authoring environment that consisted of four modules: (1) game design patterns as player interaction templates, (2) a formal analysis concerning structural errors, (3) collaborative balancing, and (4) a rapid prototyping environment.

In addition to the previously mentioned work that presented findings on game design patterns that enforce collaboration, industry experts have also discussed game mechanics and "dynamics" used to force collaboration. Specifically, Luaret[13] further defined four categories: gate, comfort, class, and job. "Gate" refers to collaboration mechanics that require all players to be present to complete a task (i.e., two players lifting a gate, hence the name). "Comfort" refers to players facing a challenge that is so difficult that having more than one player is necessary. Compared to "gate" mechanics, "comfort" mechanics indicate that it is theoretically possible but extremely difficult for a solo player to perform the given task, thus strongly encouraging collaborative behavior rather than rigidly enforcing it. Both "class" and "job" involve assigning different roles to each player, either through their player avatar or character (similar to "class") or simply through player actions (similar to "job"). Finally, Redding[14] defined several collaboration "dynamics," which describe mechanisms used to create collaborative behavior between two players. Redding placed these dynamics on a gradient from "prescriptive" (forced cooperation)

[8]https://en.wikipedia.org/wiki/List_of_cooperative_video_games
[9]https://en.wikipedia.org/wiki/Space_Duel
[10]https://en.wikipedia.org/wiki/Sky_Force
[11]https://en.wikipedia.org/wiki/Freelancer_(video_game)
[12]https://en.wikipedia.org/wiki/The_Forest_(video_game)
[13]https://www.gamasutra.com/view/news/328756/The_four_atoms_of_cooperative_video_games.php
[14]https://www.gdcvault.com/play/1014379/Keep-it-Together-Encouraging-Cooperative

to “voluntary” (encouraged but not required collaboration), which included gating/tethering, exotic challenges, punitive systems, buffing systems, asymmetric abilities, combined abilities, and survival/attrition.

However, there are also cases where developers provided practical guidelines to force collaboration in games. The developers of the *Jamestown: Legend of the Lost Colony*[15] game provided practical guidelines on designing collaborative games based on player observations[16] they made. Specifically, they suggested that game developers should “prevent waiting times,” “avoid differentiating statistics like individual scores” (which contradicts Zea et al. [82]), “take into account that the players’ skill can vary and that negative contributions could result in blaming,” “make sure that teams only fail as a collective and that each player is able to contribute something tangible,” and “facilitate interactions among the players.” Likewise, the developers of the *Together: Amna & Saif*[17] game followed similar rules to establish a relationship between the players.[18] Specifically, they included the “avoid levels that could be solved without all players contributing,” “add game mechanics that allow helping and coordination,” “have no abilities unique to each player so that each player knows exactly what the others can do” (contradicts Zagal et al. [81]), and “let players choose their responsibilities at any given time, for example to help when a player has difficulties using a certain ability.” However, we should note that these suggestions coming from research or industry sometimes differ significantly and even contradict each other in some respects. These differences highlight the fact that, in the game design process, there is no single right answer for most questions. Instead, decisions have to be made for each game individually and must be based on the intended target audience. This necessity was also pointed out by Corrigan et al. [17], who found that collaboration has to be required by the game; otherwise, the players tend to play solitarily.

In addition to collaboration in video games, the virtual reality research community has proposed various applications related to collaboration in a shared space. Zhou et al. [84] developed a collaborative asymmetrical mixed reality dance game called *Astaire*. The players of this game dance together while hitting the game targets shaped as musical notes spawning in the space. Ibayashi et al. [34] developed a collaborative experience called *Dollhouse VR*, which facilitates an asymmetric collaboration among users in and out of virtual reality. In *Dollhouse VR*, one player uses a multitouch device to interact with the virtual environment, while the other player observes and interacts with the virtual environment through a head-mounted display. Piumsomboon et al. [49] developed a remote collaborative extended reality system to create new types of collaborations across different devices. Malik et al. [43] developed a unified training tool framework to integrate human-robot interaction into a virtual reality environment. Greenwald et al. [31] developed a shared immersive virtual reality environment in which users interact to create and manipulate virtual objects by using a set of hand-based tools called *CocoVerse*. Donalek et al. [20] explored the potential of immersive visualization and data expiration in a collaborative, shared virtual space. Finally, Men and Bryan-Kinns [45] explored the potential of collaborative music-making in a shared virtual space.

Considering the abovementioned studies on collaborative games and virtual reality experiences, it is obvious that the collaborative tasks are context-dependent and diverse. Various studies have been conducted to explore how users collaborate in groups and proposed taxonomies to characterize users’ collaborative activities. For example, Tang et al. [66] identified six styles of coupling—“same problem same area,” “one working, another viewing in an engaging manner,” “same problem, different area,” “one working, another viewing,” “one working, another disengaged,” and “different problems”—where the participants were instructed to interact with a tabletop surface. Liu et al. [39] discussed five collaboration styles—Divide&Conquer (a parallel-performed task in which the users must neither communicate nor help each other), LooseComm (a parallel-performed task where the users are allowed to communicate), LooseTech (a parallel-performed task where the users can also help each other), CloseComm (only one user can perform the task in sequential order), and CloseTech (only one user can

[15]https://en.wikipedia.org/wiki/Jamestown:_Legend_of_the_Lost_Colony
[16]https://www.co-optimus.com/editorial/976/page/1/indie-ana-co-op-and-the-dev-stories-you-re-all-in-this-together.html
[17]https://togetherthegame.com/
[18]https://www.co-optimus.com/editorial/1376/page/1/indie-ana-co-op-and-the-dev-stories-fostering-gaming-relationships.html

perform the task in sequential order, but the second user also has an input device)—by operationalizing two dimensions: task parallelization and shared interaction support. The results of Liu et al. [39] study also indicated that (1) participants value collaboration even though it incurs a cost, (2) shared interaction increases collaboration, reduces physical navigation, improves operation efficiency, and provides a more enjoyable experience, and (3) distance increases the value of collaboration and shared interaction.

In the present research, we used methods such as those used in procedural content generation for virtual environments and games. Such methods, often called "constructive methods," use grammars [46, 74], noise-based algorithms [40, 75], search-based methods [42, 69], or solver-based methods [64] to generate virtual environments or game levels to maximize the objectives of the design and/or to preserve the developer-defined constraints. For example, Arkel et al. [73] introduced a platform game that utilizes a grammar-based procedural generation technique to synthesize the layout of puzzle-related game levels. Since its first successful implementation in games such as *Rogue*[19] and *Elite*,[20] procedural content generation has become a popular tool for reducing the cost of developing computer games [68]. In addition to the cost-reduction benefits, game designers can personalize games to fit players' needs and gameplay behaviors with procedural content generation techniques, leading to more personalized user experiences [49]. Procedural content generation techniques also reduce storage footprint. This was especially important in the early 1980s when memory limitations of computers and storage devices did not allow the distribution of large amounts of predesigned content, such as game levels [4, 68]. Aside from the examples mentioned above, procedural content generation in games that encounter collaborative gameplay is relatively uncommon. This is mainly because generating game levels for collaboration is more challenging due to the need to ensure the mutual benefits of the cooperation, which puts added constraints on the design spaces [73].

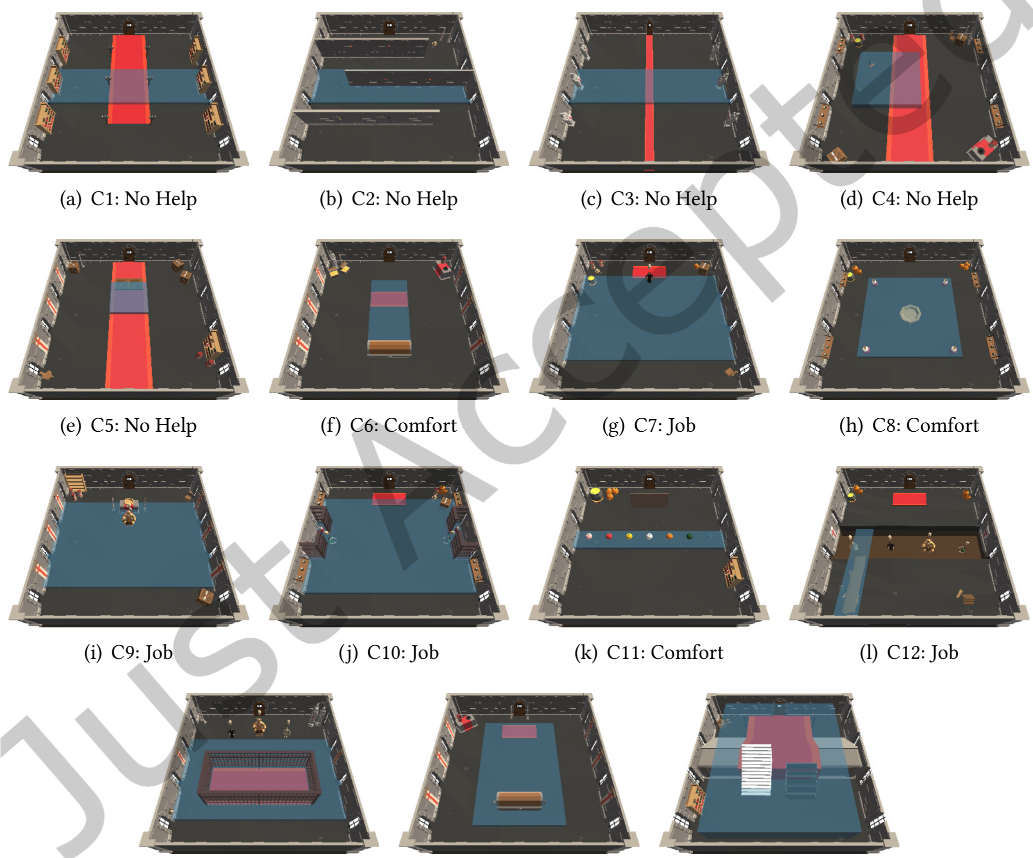
To the best of our knowledge, there are no available methods for evaluating the degree of collaboration at a game level. However, there are various previously published approaches to assessing the quality of game levels. Examples include the player challenge method [38] or the use of rapidly expanding random trees to sample a level's state space, which later clusters the output tree of the rapidly expanding random trees using Markov clustering to form a representative graph of the game level [5]. Additionally, researchers have explored spatial principles in level design to indicate the effects of altering parts of a game level [32]. Furthermore, Berseth et al. [8] used crowd simulation algorithms to evaluate the scenario complexity of game levels. In the current project, we considered the use of AI virtual agents in assessing the degree of collaboration of the designed game level chunks and, consequently, the synthesized game level; therefore, we proposed and evaluated a method to automatically determine the degree of collaboration of a synthesized game level.

For this project, we considered previously conducted research on the procedural generation of game levels and collaboration in shared virtual spaces to develop a method that automatically synthesizes game levels based on designer-specified degrees of collaboration among players and other design decisions. According to the discussed taxonomies, we mainly focused on the "same problem same area" styles of coupling between game players, as mentioned by Tang et al. [66], and in the LooseTech category of Liu et al. [39], since the players could perform a parallel task and help each other to overcome the challenges of a game level. We demonstrated that our approach can be applied to generate variations at a game level based on designer-defined objectives. Through a user study, we also validated the effectiveness of our method in generating game levels that can impact the collaborative gameplay behavior of participants.

[19]https://en.wikipedia.org/wiki/Rogue_(video_game)
[20]https://en.wikipedia.org/wiki/Elite_(video_game)

3 PRELIMINARY REMARKS

In this section, we present the different game level chunks developed for our project and the methods we followed to characterize the degrees of collaboration for each game level chunk. We considered synthesizing game levels for this project's obstacle course game. Our system composes a game level by placing game level chunks next to each other in a 1D array structure. We chose a simplified representation of a game level mainly to validate whether the presented methodology can synthesize game levels that fulfill the degree of collaboration targets and other design decisions. In addition, through our user study, we aimed to explore whether the participants could play the synthesized game levels and experience a certain degree of collaboration for each other. Thus, we leave more complex game level structures (e.g., dungeon crawlers and open-world game levels) for future implementations.



(a) C1: No Help (b) C2: No Help (c) C3: No Help (d) C4: No Help

(e) C5: No Help (f) C6: Comfort (g) C7: Job (h) C8: Comfort

(i) C9: Job (j) C10: Job (k) C11: Comfort (l) C12: Job

(m) C13: Job (n) C14: Gate (o) C15: Gate

Fig. 2. Playable game level chunks were developed by an experienced game level designer and used in this project to synthesize game levels and account for the degrees of collaboration. We also characterized each game level chunk based on Luaret's taxonomy. The blue shapes indicate the collaboration zones of each game level chunk.

3.1 Game Level Chunks

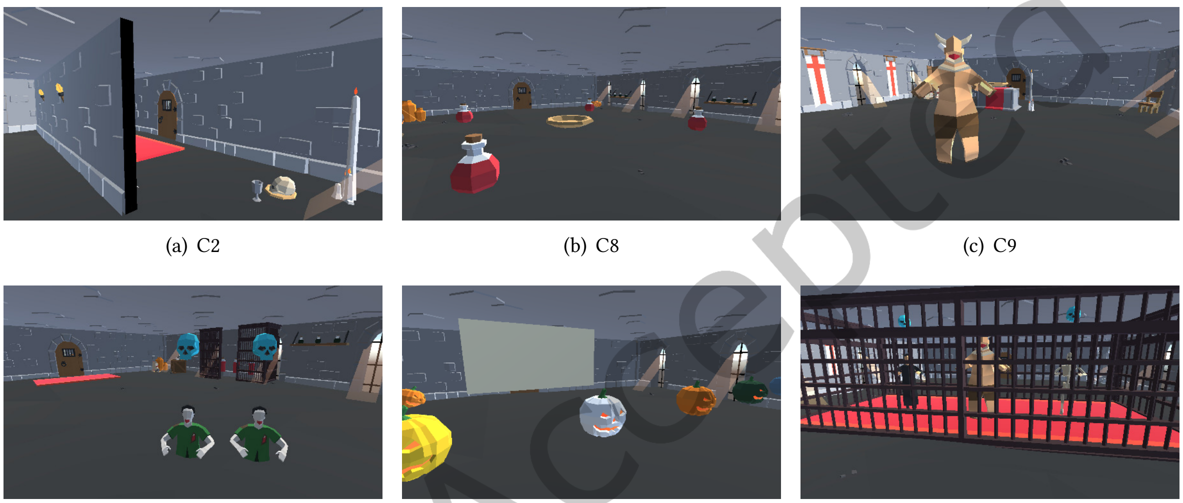
In a preliminary step, we asked an experienced game level designer to design playable game level chunks, considering different collaboration activities and the different degrees of collaboration players need to finish each game level chunk. The designer created 15 game level chunks. Fig. 2 illustrates all game level chunks, where "playable game level chunks" denotes a part of the game that has its own gameplay characteristics and objectives and is independent of the other game level chunks.

Based on the theories of designing collaborative gameplay by Rocha et al. [53], Luaret,[13] and Redding,[14] game level chunks can be divided into three categories: (1) chunks that a player can complete on their own without the help of another player (C1, C2, C3, C4, and C5); (2) chunks that a player can complete without the help of another player—however, if another player helps, the players will complete the chunk faster (C6, C7, C8, C9, C10, C11, and C12); and (3) chunks that if players do not collaborate to complete, they will become "stuck" and not be able to exit the chunk (C13, C14, and C15). Each of these chunks are described as follows:

- **C1:** The exit door of this game level chunk opens when a player enters the room.
- **C2:** This is a simple maze where no collaboration is required. Once a player reaches the red zone, the exit door of this game level chunk opens.
- **C3:** The players cannot pass the narrow exit door simultaneously. Its exit door opens when a player enters the room.
- **C4:** A player should touch the pumpkin to open the exit door of this game level chunk.
- **C5:** There is a large button on the floor in this game level chunk. Its exit door opens once a player jumps on the button.
- **C6:** The player(s) should push the chest to move it to a specific place (red zone). The speed of the chest increases proportionally to the number of players pushing it. The exit door opens only when the player(s) places the chest on the red zone.
- **C7:** One player should attract the enemy's attention while the other player reaches the red zone to open the exit door of this game level chunk. In the case of a single player, that player should feint the enemy to reach the red zone to open the exit door.
- **C8:** In this game level chunk, there are four bottles. The player(s) should grab the bottles and put them in the basket. Once all bottles are in the basket, the exit door of this game level chunk opens.
- **C9:** There is a scroll attached to the back of the enemy. The players should collaborate to "steal" the scroll. In particular, one player should attract the enemy's attention, while the other player "steals" the scroll. When a player places the scroll in the basket, the exit door of this game level chunk opens. In the case of a single player, that player should feint the enemy to "steal" the scroll.
- **C10:** One player should collect the bottles and place them in a designated position, while the other player should attract the enemies. When the players have placed all bottles in the designated position (wooden baskets), the exit door of this game level chunk opens. In the case of a single player, that player should run fast to prevent the enemy from collecting the bottles and placing them in a designated position.
- **C11:** The player(s) need to touch the pumpkins according to a particular color sequence shown on a board to open the exit door of this game level chunk. If the players collaborate, they will be able to exit this room faster.
- **C12:** A player must carry the board and place it in a suitable place to form a bridge. When a player reaches the red zone, the exit door of this game level chunk opens.
- **C13:** In this game level chunk, players can open and close a cage by touching a button. One player is responsible for controlling the cage, while the other is responsible for directing the enemies to the cage. Only once the players trap all enemies in the cage does the exit door of this game level chunk open.

- **C14:** The players should grab the chest together and move it to the designated place (red zone) to open the exit door of this game level chunk.
- **C15:** Once a player reaches the top of the wall using the black ladder, the ladder breaks. The player should then push the white ladder down to allow the other player to climb the wall. When a player reaches the red zone, the exit door of this game level chunk opens. If the first player that reaches the top does not push down the white ladder, the second player will become "stuck" and not be able to exit this chunk.

Fig. 3 illustrates different game level chunks from a first-person perspective. Moreover, we provide gameplay examples of the synthesized game levels in the accompanying video. All game levels and our implementations can be found on our project's website and downloaded from there.



(a) C2 (b) C8 (c) C9

(d) C10 (e) C11 (f) C13

Fig. 3. Example scenes of the developed game level chunks from a first-person perspective.

3.2 Game Level Chunk Characterization

Our characterization process begins by specifying the collaboration zones at each game level chunk. We adopted the idea of using collaboration zones from Reuter et al. [52], who described various patterns that enforce collaboration between players. In the current project, the collaboration zones are designer-specified areas inside the game level chunks in which we expect both players to be present simultaneously; this means that the players collaborate to accomplish each given task. Fig. 2 illustrates the collaboration zones of different game level chunks.

For example, in the case of the C6 game level chunk (Fig. 2(f)), the players should push the chest to move it to the designated position to open the exit door. The collaboration zone of this chunk covers the path that the players should follow when pushing the chest to the designated red zone. Thus, if both players are present in this collaboration zone and try to push the chest together, a high degree of collaboration will characterize that game level chunk. Therefore, the players can push the chest faster and consequently exit that game level chunk more quickly. In this paper, we define the degree of collaboration as the time ratio for which the virtual avatars are inside the collaboration zone of a game level chunk over the total time spent in that game level chunk, which, in practice, can be translated as the "same problem same area," as defined by Tang et al. [66].

According to the literature [41, 80], the designer who created the game level chunks could have characterized the degree of collaboration of game level chunks, or we could have recruited participants to play each game level chunk and capture the necessary data to characterize each of them. However, building on these approaches and adopting the ideas of Berseth et al. [6], we used AI virtual agents to play each game level chunk. We did so because, first, the AI virtual agents could provide more accurate data on the exact degree of collaboration required to complete each game level chunk. Second, we aimed to explore the potential of using AI virtual agents as an alternative method for evaluating the degree of collaboration of a game level chunk and, consequently, of a game level. We also decided to use AI virtual agents, as several previous studies have proved that the use of AI (virtual) agents for playtesting can provide reasonable playtesting data [19, 27, 29]. In our pipeline, we integrated AI virtual agents that repeated the gameplay of each game level chunk at super-speed in a headless mode. In addition, we introduced some variations in the simulation (e.g., changing the starting position of each AI virtual agent) to capture variations in how the AI virtual agents could play each game level chunk. Thus, although we considered that each trial of the AI virtual agents might prove less useful than human data within a fixed budget or time, the proposed automatic method could create more data.

For our AI virtual agents, we first developed behavior trees similar to those developed by Shoulson et al. [61] with a set of tasks in a modular fashion that our system could use to allow the AI virtual agents to play and exit each game level chunk successfully. Given the behavior tree that corresponds to a given game level chunk, the AI virtual agents selected and executed the most appropriate interaction and collaboration pattern during the runtime of the gameplay. In the Appendix of this paper, we present the behavior trees we developed for the different game level chunks and, consequently, for the different behaviors assigned to the developed AI virtual agents.

To obtain the degree of collaboration of each game level chunk, we assigned a random position to each AI virtual agent at the entrance of each game level chunk and captured the degree of collaboration that characterized a given game level chunk. For each game level chunk, we repeated this process 10 times by randomizing the initial position of each AI virtual agent at the beginning of their gameplay. Then, at each game level chunk, we assigned the average degree of collaboration of the 10 trials as the value that characterizes that particular game level chunk. As mentioned, we denote the ratio between the time the AI virtual agents spent inside the collaboration zone of a game level chunk to the total time spent in that game level chunk as the degree of collaboration. Table 1 lists the obtained values characterizing the degree of collaboration of each game level chunk.

4 PROBLEM FORMULATION AND OPTIMIZATION

Our approach synthesizes game levels with respect to the degree of collaboration and other design decisions. We outline a detailed description of the problem formulation and optimization in the following subsections.

4.1 Formulation

We begin by denoting a game level (L) composed of a designer-defined number of game level chunks (c_i) assembled in a sequential order. We represent the synthesis of the game level (L) with a total cost function (C_{Total}) that encodes our game level design considerations:

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

Here, $\mathbf{C}_{\text{Collab}} = [C_{\text{Collab}}^M, C_{\text{Collab}}^V, C_{\text{Collab}}^P]$ is a vector of collaboration-related costs, and $\mathbf{w}_{\text{Collab}} = [w_{\text{Collab}}^M, w_{\text{Collab}}^V, w_{\text{Collab}}^P]$ is a vector of the corresponding weights, where each weight $\in [0, 1]$. C_{Collab}^M, C_{Collab}^V, and C_{Collab}^P encode the collaboration-related design decisions: the mean degree of collaboration required to complete the synthesized game level, the variation in the degree of collaboration, and the progress of the degree of collaboration across the game level chunks. $\mathbf{C}_{\text{Prior}} = [C_{\text{Prior}}^S, C_{\text{Prior}}^R]$ is a vector of game level prior costs that encodes design decisions, such

Table 1. Classification of the game level chunks based on Luaret's taxonomy, the degree of collaboration of each game level chunk based on the data obtained from the AI virtual agents, the percentage of the collaboration zone over the total area of the game level chunk, and the category to which each chunk belongs (* chunks that a player can complete on their own without the help of another player; ** chunks that a player can complete without the help of another player—however, if another player helps, the players will complete the chunk faster; and *** chunks that if players do not collaborate to complete, they will become "stuck" and will not be able to exit the chunk).

Chunk ID	Luaret's Taxonomy	$D(c_i)$	Collaboration Zone (%)	Chunk Category
C1	N/A	.21659	25.00	*
C2	N/A	.21131	25.00	*
C3	N/A	.21744	25.00	*
C4	N/A	.32782	14.00	*
C5	Job	.27382	6.25	*
C6	Comfort	.51531	13.43	**
C7	Job	.49580	62.50	**
C8	Comfort	.52015	34.51	**
C9	Job	.45949	62.50	**
C10	Job	.70475	68.75	**
C11	Comfort	.40382	12.50	**
C12	Job	.43350	12.58	**
C13	Job	.77391	56.25	***
C14	Gate	.71462	37.50	***
C15	Gate	.76937	65.63	***

as the size of the game level (number of game level chunks) and repetition among adjacent game level chunks. As mentioned before, $\mathbf{w}_{\text{Prior}} = [w^S_{\text{Prior}}, w^R_{\text{Prior}}]$ is a vector of the corresponding weights, where each weight $\in [0, 1]$. Based on the above formulation, we provide the game developers with the ability to control the design decisions related to the game level by changing the target of each cost term. In addition, we provide them with the ability to control the output synthesized game levels by allowing them to change the priority (weight) of each cost term. This means that even if the game level designer sets a target value for a specific cost term, if the assigned weight of that cost term is a low value, such a design decision might not appear in the synthesized game level due to its low priority. In contrast, if a designer assigns a high weight value to a cost term, such a design decision would appear at the synthesized game level.

4.2 Collaboration Costs

We developed three cost terms to encode the design decisions regarding the degree of collaboration at a game level (L). The collaboration costs include the mean degree of collaboration, variation in the degree of collaboration, and progress in the degree of collaboration.

Mean Degree of Collaboration Cost: We define a cost term to control the mean degree of collaboration the game players require to accomplish the game level (L). We define this cost as follows:

$$C^M_{\text{Collab}}(L) = \left(\frac{1}{|L|} \sum_{c_i} \mathcal{D}(c_i) - \rho_M \right)^2, \quad (2)$$

where $\rho_M \in [0, 1]$ is the target mean degree of collaboration, and $\mathcal{D}(c_i)$ is the degree of collaboration of the (c_i) game level chunk. By assigning a low ρ_M value to the above equation, our system will synthesize a game level in which the users will expect low collaboration to finish that game level, while by assigning a high ρ_M target value, the system will most likely synthesize a game level that the users will not be able to finish without collaboration. Fig. 4 illustrates the game levels synthesized by varying the value of ρ_M.

Variation in the Degree of Collaboration Cost: We define a variation in the degree of collaboration cost to consider the range of the collaboration required among the selected game level chunks, as follows:

$$C^{V}_{\text{Collab}}(L) = \left| \frac{1}{|L|} \sum_{c_i} \left(\mathcal{D}(c_i) - \bar{\mathcal{D}} \right)^2 - \rho_V \right|, \tag{3}$$

where $\rho_V \in [0, 1]$ is the target variation in the degree of collaboration, and $\bar{\mathcal{D}}$ is the mean of the degree of collaboration of the game level chunks. By changing the ρ_V target value, the developer can specify the variation in the degree of collaboration at the synthesized game level. In particular, by assigning a low ρ_V, the synthesized game level will comprise game level chunks whose degree of collaboration is close to the mean degree of collaboration target (ρ_M), while when the ρ_V target value is high, we will observe in the synthesized game level, game level chunks from the whole spectrum of the degree of collaboration we have in our dataset.

Degree of Collaboration Progress Cost: This cost controls the progression of the degree of collaboration along the synthesized game level. For this purpose, we allow the developer to define a line graph (G) with a number ($|L|$; equal to the size of the level) of elements (g_i; each g_i corresponding to a target degree of collaboration value). This line graph is used as a reference to synthesize a game level with a degree of collaboration across the game level chunks comprising L and aligning with the designer-defined line graph (G) while following the designer-defined mean collaboration cost. We define the degree of collaboration progress cost as follows:

$$C^{P}_{\text{Collab}}(L) = \frac{1}{|L|} \sum_{c_i} \Big(\mathcal{N}\big(\mathcal{D}(c_i)\big) - \mathcal{N}\big(\mathcal{D}(g_i)\big) \Big)^2, \tag{4}$$

where g_i is the target degree of collaboration for the $i-th$ game level chunk from the pre-defined line graph. $\mathcal{N}$ denotes the normalized values of the degree of collaboration, $\mathcal{D}(c_i)$, of the game level chunk (c_i) of the game level (L) and the target degree of collaboration, $\mathcal{D}(g_i)$, of the element (g_i) of the input line graph (G). A designer can easily control the progress of the degree of collaboration by choosing from a list of predefined curves and lines (we illustrate line graphs and the corresponding game levels in Fig. 5) or by defining and importing a new progression line graph (G). Based on this functionality, the game level designer can specify the targets of the mean degree of collaboration (ρ_M) and variance of the degree of collaboration (ρ_V). Then, the line graph specifies the progression of the game level chunks across the systemized game level. This functionality provides the game level designer with additional control over the synthesis process of a game level.

4.3 Prior Costs

We define the prior cost terms to encode specific game level design decisions. Among other variables, we choose the size (number of game level chunks) that constitutes a game level and the repetition of adjacent game level chunks.

Size Cost: We define a level size cost for constraining the number of game level chunks that compose a game level, as follows:

$$C^{S}_{\text{Prior}}(L) = 1 - \exp\left(- \frac{1}{2\sigma_S^2} \big(|L| - \rho_S\big)^2 \right), \tag{5}$$

where ρ_S is the designer-defined number of game level chunks, and σ_S controls the spread of the Gaussian penalty function, which is empirically set as $\sigma_S = 1.00$.

Adjacent Repetition Cost: We also define a cost to penalize the repartition of similar game level chunks, therefore eliminating the synthesis of monotonic game levels in which similar game level chunks are placed next to one another. We represent the adjacent repetition cost as follows:

$$C^{R}_{\text{Prior}}(L) = \frac{1}{|L|-1} \sum_{c_i, c_{i+1}} \Gamma(c_i, c_{i+1}), \tag{6}$$

where c_i and c_{i+1} are adjacent game level chunks in L, and $\Gamma(c_i, c_{i+1})$ returns a high value if c_i and c_{i+1} are identical and a low value otherwise, under following the condition:

$$\Gamma(c_i, c_{i+1}) = \begin{cases} 1 & \text{if } (c_i \equiv c_{i+1}) \\ 0 & \text{otherwise} \end{cases}.$$

In conclusion, game developers can consider various other prior costs depending on the game's objectives and design decisions.

4.4 Optimization

Given the game level designer-defined decisions, our system optimizes the total cost function by applying a Markov-chain Monte Carlo (MCMC) [30] method, known as "simulated annealing," with a Metropolis-Hastings [13] state-searching step. Given that any number of game level chunks can synthesize a game level, a trans-dimensional solution space encodes all possible design outcomes of a game level. Thus, to successfully sample the solution spaces of game levels assembled by several game level chunks, we use the reversible-jump [21] variation in the MCMC technique. For our optimization process, we start by defining a Boltzmann-like objective function:

$$f(L) = \exp\left(-\frac{1}{t} C_{\text{Total}}(L)\right), \tag{7}$$

where t encodes the temperature parameter of simulated annealing. Given the current game level (L) during the optimization process, the optimizer proposes a change to that game level, creating a proposed game level (L'). In particular, to obtain the proposed game level (L'), our system updates the current game level (L) by choosing one of the following moves:

- **Add a Game Level Chunk:** When this move is selected, the system randomly selects a game level chunk from our game level chunk set and places it in a randomly chosen location within the game level.
- **Remove a Game Level Chunk:** In this move, the system randomly selects a game level chunk from the current layout (L) and removes it.
- **Replace a Game Level Chunk:** In this move, from the current game level, the system randomly selects a game level chunk from the current layout (L) and replaces it with a randomly selected game level chunk from our game level chunk set.

In our method, we set the probabilities of "add a game level chunk" as $p_{\text{add}} = .40$, "remove a game level chunk" as $p_{\text{remove}} = .20$, and "replace a game level chunk" as $p_{\text{replace}} = .40$. This approach selects the "add a game level chunk" and "replace a game level chunk" moves with higher probability.

The optimizer accepts a proposed game level configuration (L') by comparing its total cost value, $C_{\text{Total}}(L')$, with the total cost value, $C_{\text{Total}}(L)$, of the current layout (L). To ensure a detailed balanced condition in trans-dimensional optimization, the optimizer accepts a proposed layout (L') based on the acceptance probabilities for the "add a game level chunk," "remove a game level chunk," and "replace a game level chunk" moves. We define the probability of the "add a game level chunk" move as:

$$p_{\text{add}}(L'|L) = \min\left(1, \frac{p_{\text{remove}}}{p_{\text{add}}} \frac{U - |L|}{|L'|} \frac{f(L')}{f(L)}\right); \tag{8}$$

the probability for the "remove a game level chunk" move as:

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

and the probability for the "replace a game level chunk" move as:

$$p_{\text{replace}}(L'|L) = \min\left(1, \frac{f(L')}{f(L)}\right). \tag{10}$$

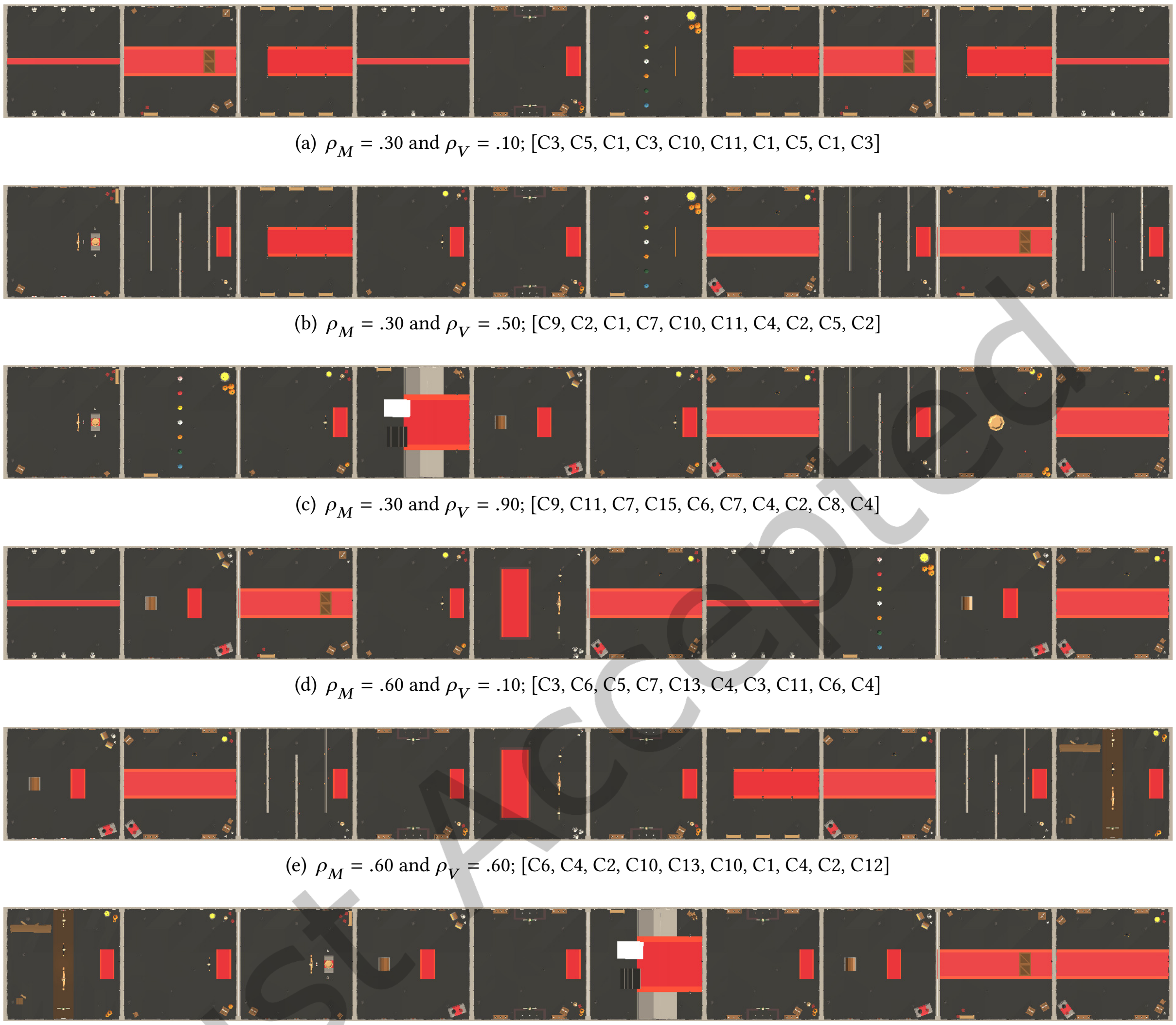
The acceptance probabilities during the optimization process consider the variable U, which denotes the upper limit of the number of game level chunks. For formulation simplicity, we assume that each game level chunk (c_i) can only be selected (U_i) times rather than an infinite number of times. Thus, our system synthesizes a level of up to $U = \sum_i U_i$ game level chunks. In our implementation, we set $U = 20$ for all game level chunks.

We implement simulated annealing to effectively explore the solution space. Regarding the temperature parameter (t) of the optimizer, at the beginning of the optimization, we set t to a high value such that the optimizer aggressively explores the whole solution space, decreasing gradually until reaching a value near zero. We initialize the temperature as $t = 1.00$ at the beginning of the optimization and multiply it by $t^* = .998$ after each iteration. The optimizer becomes "greedier" when refining the optimal solution as the iteration evolves. The optimization terminates when the change in $C_{\text{Total}}(L)$ is less than 2.5% over the last 50 iterations.

Unless we specify otherwise, for all collaboration-related cost terms presented in this paper, we set the weights at $w^M_{\text{Collab}} = 1.00$, $w^V_{\text{Collab}} = .30$, and $w^P_{\text{Collab}} = .50$. For the prior cost terms, we set the weights at $w^S_{\text{Prior}} = 1.00$ and $w^R_{\text{Prior}} = .50$. We assign a high weight value to w^M_{Collab} as we want the optimizer to prioritize the corresponding cost term and synthesize a game level whose mean degree of collaboration is as close as possible to the designer-specified target value ρ_M. In addition, we assign a high value to w^S_{Prior} as we want our system to synthesize a game level whose size is the requested one. If, for example, we assign a lower value to w^S_{Prior}, our system might compose a game level with either less or more game level chunks since the system would have first tried to fulfill the design decisions having higher weight values and, consequently, higher priorities than those with lower weight values. Finally, we assign low and medium values to w^V_{Collab}, w^P_{Collab}, and w^R_{Prior} as such design decisions should not be prioritized by the optimizer. The designer can also control the priority of each design goal at a given game level by changing these weights. Fig. 4 illustrates the examples of the synthesized game levels with different targets for the collaboration cost terms. Fig. 5 shows the game levels synthesized using various degrees of collaboration progress line graphs while keeping the mean degree of collaboration target and variation in the degree of collaboration constant.

5 USER STUDY

In this study, we explored whether our developed method can synthesize game levels with different targeted degrees of collaboration, thereby impacting the participants' gameplay behavior. Moreover, we attempted to evaluate whether the AI virtual agents can characterize the degree of collaboration in the game level chunks. We provide more details about the study and our results in the following sections.



(a) ρ_M = .30 and ρ_V = .10; [C3, C5, C1, C3, C10, C11, C1, C5, C1, C3]

(b) ρ_M = .30 and ρ_V = .50; [C9, C2, C1, C7, C10, C11, C4, C2, C5, C2]

(c) ρ_M = .30 and ρ_V = .90; [C9, C11, C7, C15, C6, C7, C4, C2, C8, C4]

(d) ρ_M = .60 and ρ_V = .10; [C3, C6, C5, C7, C13, C4, C3, C11, C6, C4]

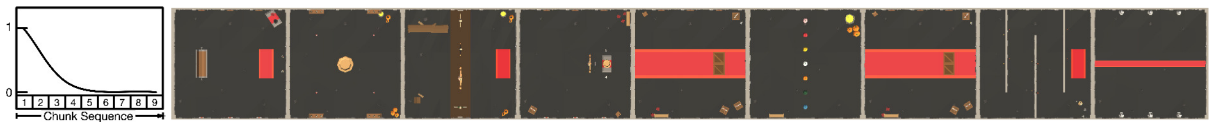
(e) ρ_M = .60 and ρ_V = .60; [C6, C4, C2, C10, C13, C10, C1, C4, C2, C12]

(f) ρ_M = .60 and ρ_V = .90; [C12, C7, C9, C6, C10, C15, C10, C6, C5, C4]

Fig. 4. Different game levels synthesized by our system by varying the targets of our cost terms. For all examples, we set the weights of the collaboration-related cost terms at w^M_{Collab} = 1.00, w^V_{Collab} = .30, and w^P_{Collab} = .50, and those of the prior cost terms at w^S_{Prior} = 1.00 and w^R_{Prior} = .50. The same game level chunk can appear more than once at a synthesized level (e.g., C1, C3, and C5 in Fig. 4(a)); however, due to the adjacent repetition cost term, the system does not repeat the same chunk one after the other.

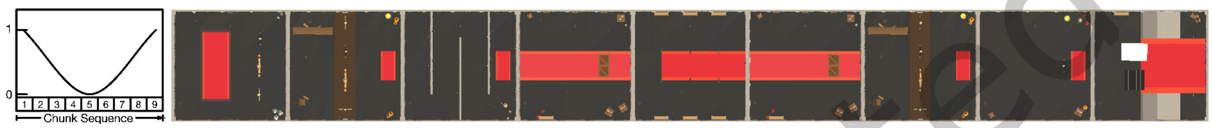
5.1 Participants

We conducted an *a priori* power analysis [15] to determine the sample size for our study, using the G*Power version 3.10 software [23]. The calculation was based on one group with three repeated measures, 90% power,

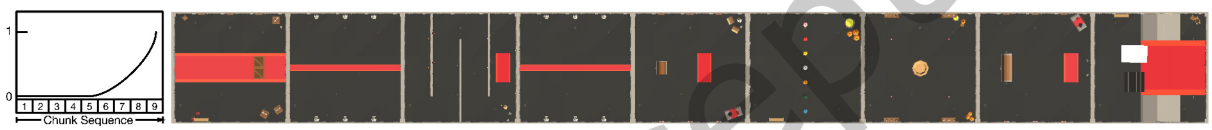


(a) [C14, C8, C12, C9, C5, C11, C5, C2, C3]

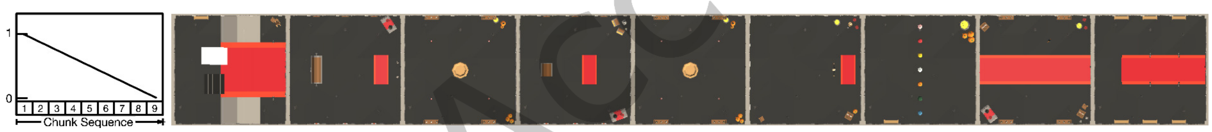
(b) [C2, C1, C4, C11, C10, C13, C11, C4, C1]



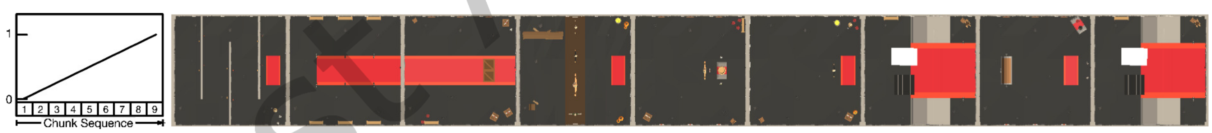
(c) [C13, C12, C2, C5, C1, C5, C12, C7, C15]



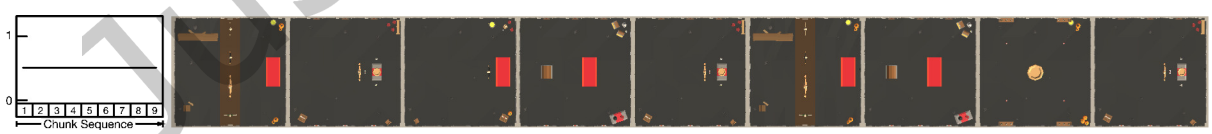
(d) [C5, C3, C2, C3, C6, C11, C8, C14, C15]



(e) [C15, C14, C8, C6, C8, C7, C11, C4, C1]



(f) [C2, C1, C5, C12, C9, C7, C15, C14, C15]



(g) [C12, C9, C7, C6, C9, C12, C6, C8, C9]

Fig. 5. Example game levels (ρ_S = 9) using different degrees of collaboration progress line graphs while maintaining the mean degree of collaboration target constant. For all examples, we use ρ_M = .50 and ρ_V = .50 as the targets.

medium-to-large effect size of f = .35 [22], non-sphericity correction ϵ = .70, correlation among repeated measures of $r = .50$, and $\alpha = .05$. The analysis resulted in a recommended sample size of 25 groups of participants (for clarification, each group was composed of two students).

We recruited the participants through e-mails sent to our department's undergraduate and graduate students. As we conducted this study to explore the collaborative behavior of our participants during gameplay, they were scheduled to attend the sessions in groups of two. In total, 50 students participated in our study (25 groups of students). The age range of our participants was $18 - 29$ years (age: $M = 19.28$, $SD = 1.79$). All participants had previously experienced virtual reality, and all of them played video games regularly. The participants in each group were randomly assigned to minimize the chances that the groups were composed of students who knew each other. The research team also asked a designated question before the beginning of the study. Our results indicated that no group was composed of students who had played games together in the past. We did not provide monetary compensation to our participants for their participation; however, we provided snacks and water to them throughout the study session to compensate them for their time and effort.

5.2 Setup and Implementation Details

This study was conducted in a laboratory in our department. We used the Unity game engine version 2019.4.12 to develop our application and ran the application on two (one computer per participant) Dell Alienware Aurora R7 desktop computers (Intel Core i7, NVIDIA GeForce RTX 2080, 32GB RAM). The optimization of the game level with $\rho_S = 10$ game level chunks did not exceed five seconds. We used Oculus Quest and its Unity SDKs (Oculus Integration). Finally, we used the Photon Unity Networking[21] asset to enable the networking functionality between the two computers and, consequently, to allow the participants to collaborate in a shared virtual space.

5.3 Experimental Conditions

We developed three experimental conditions (game levels) to determine whether optimizing the game levels with different targeted degrees of collaboration would impact the collaboration gameplay behavior of our participants. We followed a within-group study design, which meant that all participant groups played the three developed game levels. To balance the conditions across the participant groups and minimize the carryover effect of gameplay knowledge across game levels with different degrees of collaboration targets, we used the Latin squares [36] ordering method. We used $\rho_S = 10$ as the target size of the game levels for all three conditions. The conditions were as follows:

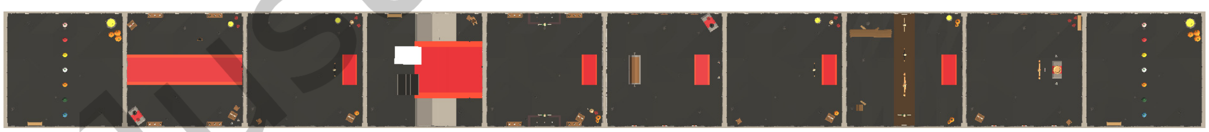
- **Low Collaboration (LC):** We requested that our system create an LC game level expecting that our participants could finish it with minimal to no collaboration necessary. We set the target value of the degree of collaboration cost term at $\rho_M = .30$. Under this condition, we expected the synthesized game level to be composed mainly of the game level chunks that require low to medium degree of collaboration activity (C1-C12).
- **Medium Collaboration (MC):** Under this condition, we requested that our system synthesize a game level in which our participants would moderately collaborate to finish it. This meant that if the participants collaborated on some parts of the game level, they would complete the game faster. We set $\rho_M = .50$. Under this condition, we expected the synthesized game level to be composed of game level chunks from the whole spectrum of the degree of collaboration (C1-C15).
- **High Collaboration (HC):** Under the last condition, we requested our system to synthesize a game level in which the participants should collaborate even more to finish the level. We set $\rho_M = .70$. In HC, it is highly likely that if the participants do not collaborate, they will not be able to finish the game. Under this condition, we expected the synthesized game level to be composed of game level chunks that require medium to high collaboration activity (C6-C15).

[21]https://www.photonengine.com/pun

We did not change the weights assigned to collaboration and prior costs across the experimental conditions. However, we set a different target value to the mean degree of collaboration cost term; therefore, we requested our method to synthesize a game level with a certain goal (i.e., a different degree of collaboration target). Additionally, for the degree of collaboration progress term, we used a Gaussian-like line graph as a reference (similar to Fig. 5(b)). This meant that the system should synthesize the game level for which at the start and end of a level, we would be able to observe game level chunks of low degree of collaboration. In contrast, we would observe game level chunks of a higher degree of collaboration in the middle of the game level. We synthesized our game levels in such a way for three reasons. First, we did not want to synthesize monotonic game levels with a near-equal degree of collaboration across the game level chunks. Second, we wanted to synthesize game levels that included game level chunks of low amd medium degree of collaboration activity, similar to most commercial games (i.e., most games have designated areas at each game level that require more collaboration than other areas at the same level). Third, during a preliminary study, we realized that when we placed higher collaboration game level chunks toward the end of the synthesized game level, the participants tended to collaborate more than they actually collaborated. This indicated that the participants' collaborative gameplay experiences at the end of game levels tended to override those at the beginning of the same game levels. Fig. 6 shows the three synthesized game levels we used in our study. The LC game level (Fig. 6(a)) indicated that such a game level is mainly composed of low collaboration activity game level chunks, the MC game level (Fig. 6(b)) is primarily formed by medium collaboration activity game level chunks, and the HC game level (Fig. 6(c)) is mainly composed of medium and high collaboration activity game level chunks.

(a) $\rho_M = 0.30$ and $\rho_V = 0.50$; [C2, C5, C3, C5, C11, C2, C1, C3, C4, C2]

(b) $\rho_M = 0.50$ and $\rho_V = 0.50$; [C9, C2, C1, C12, C8, C7, C8, C6, C9, C13]



(c) $\rho_M = 0.70$ and $\rho_V = 0.50$; [C11, C4, C7, C15, C10, C14, C7, C12, C9, C11]

Fig. 6. Three different synthesized game levels used in our study. From top to bottom: (a) low degree of collaboration, (b) medium degree of collaboration, and (c) high degree of collaboration.

5.4 Measurements

For our study, we collected both objective and subjective data. We collected the **degree of collaboration** regarding objective data mainly to understand how the three different conditions impacted the two participants when playing at the synthesized game levels. However, we also performed several other in-game measurements to evaluate the potential use of AI virtual agents as a method for assessing the degree of collaboration at the game level. In particular, we collected the following data:

- **Degree of Collaboration:** The ratio of time for which the virtual avatars were inside the collaboration zone to the total time spent at the game level.
- **Player Distance:** The average distance between two virtual avatars during gameplay.
- **Travel Distance:** The average length of the trajectory that the two virtual avatars traveled in the game.
- **Completion Time:** The total time players spent finishing the game (the timer stopped when the second player finished the game).
- **Collaboration Time:** The total time for which the virtual avatars were inside the defined collaboration zones.
- **Close Proximity Time:** The total time for which the two virtual avatars were in close proximity to each other (inside one another's personal space).

In addition to the objective data, we collected subjective data based on a scale we developed. Inspired by Thomson et al. [67] empirically validated theory of collaboration, we created a **perceived collaboration** scale comprising six items (Table 2) to capture how the participants perceived the degrees of collaboration at the synthesized game levels. We collected the responses from our participants using a seven-point Likert scale, where 1 = "not at all" and 7 = "totally."

Table 2. Perceived collaboration scale used in this study.

Label	Statement
Q1	During the gameplay, I felt I belonged to the group.
Q2	During the gameplay, I felt I helped the group.
Q3	During the gameplay, I felt I helped my partner.
Q4	During the gameplay, I felt my partner was helping me.
Q5	During the gameplay, a collaborative atmosphere was created.
Q6	During the gameplay, I collaborated with my partner to finish the game.

5.5 Procedure

After scheduling a date and time with the research team, the participants arrived at the laboratory in our department. Upon arrival, the researchers provided the participants with informed consent forms approved by the university's Institutional Review Board. The participants were required to sign up for inclusion in the study. Next, the research team instructed the participants to provide their demographic information by filling out the questionnaire. Once both participants from each group were in the laboratory, the research team helped them with the virtual reality equipment.

The research team was responsible for starting the game using the desktop computer. The research team instructed the participants to play a game composed of different game level chunks. Before the game started, we provided a short tutorial to all participants to familiarize them with the controllers. A previous study showed that such tutorials can improve participants' performance and player experience [35]. When the research team clicked the play button in Unity, the participants first saw the game level. Both participants were in the same shared real environment (our laboratory space) and virtual space (Fig. 1). Once the game began, the research team instructed the participants to play the synthesized game level, with the goal of finishing the game level. The research team did not provide further information to the participants about the game and gameplay. They also did not tell the participants whether they would need to collaborate with their partner during gameplay. They were left to explore on their own whether such collaboration would be necessary. The research team informed the participants that an on-screen indicator would notify them when they finished the game level.

The researchers were responsible for setting up each subsequent game level. After the end of each game level (see Fig. 6 for the LC, MC, and HC game levels), the participants were instructed to self-report their perceived collaboration (Table 2) through Qualtrics, which is a web-based survey tool provided by our university. We allowed the participants to take a short break between the experimental conditions. No participant group spent more than 60 min completing the study. We also told the participants that they could quit the study at any time; however, no team quit the study.

5.6 Results

We used a one-way repeated measures analysis of variance to explore potential differences across the examined conditions. We evaluated the normality of the collected data using Shapiro-Wilk tests to the 5% level and the residuals' graphic Q-Q plots. The Shapiro-Wilk tests and Q-Q plots indicated that our data were normal. Moreover, we screened the internal validity of the perceived collaboration scale using Cronbach's alpha coefficient. With sufficient scores ($\alpha = .81$ for the LC game level, $\alpha = .89$ for the MC game level, and $\alpha = .77$ for the HC game level), we used a cumulative score for the six items. The removal of items would not have enhanced these reliability measures. We used a p-value of $< .05$ to denote statistical significance. Finally, we used Bonferroni-corrected estimates for our post-hoc comparisons.

5.6.1 In-game Measurements. Table 3 shows the descriptive statistics for the in-game measurements. The analysis of the **player distance** data did not reveal any significant results ($\Lambda = .770$, $F[2, 23] = 3.442$, $p = .526$, $\eta_p^2 = .019$). Similarly, the **close proximity time** measurement data did not reveal any statistically significant differences ($\Lambda = .762$, $F[2, 23] = 3.589$, $p = .349$, $\eta_p^2 = .039$) across the examined conditions.

The analysis of the **degree of collaboration** measurement revealed significant differences across the examined conditions ($\Lambda = .065$, $F[2, 23] = 166.730$, $p = .0001$, $\eta_p^2 = .935$). The results of post-hoc analysis revealed that the degree of collaboration during the LC condition ($M = .17$, $SD = .06$) was significantly lower than that during the MC condition ($M = .40$, $SD = .03$), at $p = .001$, and the HC condition ($M = .45$, $SD = .04$), at $p = .0001$. Moreover, the degree of collaboration during the MC condition was significantly lower than that during the HC condition, at $p = .001$.

We identified significant results for the **travel distance** measurement ($\Lambda = .095$, $F[2, 23] = 109.548$, $p = .0001$, $\eta_p^2 = .905$). The results of the post-hoc analysis revealed that the participants in the LC condition ($M = 642.69$, $SD = 36.90$) traveled less than that in the MC condition ($M = 717.40$, $SD = 58.20$), at $p = .001$, and the HC condition ($M = 799.19$, $SD = 93.41$), at $p = .0001$. Moreover, the participants in the MC condition traveled less than they did in the HC condition, at $p = .007$.

The **completion time** measurement was also statistically significant [$\Lambda = .091$, $F(2, 23) = 115.385$, $p = .0001$, $\eta_p^2 = .909$]. The results of the post-hoc analysis revealed that the participants in the LC condition ($M = 110.73$, $SD = 16.54$) spent less time finishing the game than that in the MC condition ($M = 146.15$, $SD = 24.61$), at $p = .001$, and the HC condition ($M = 178.91$, $SD = 31.70$), at $p = .001$. Moreover, the time that the participants spent finishing the MC condition was significantly lower than that in the HC condition, at $p = .002$.

Finally, the **collaboration time** measurement was also statistically significant [$\Lambda = .048$, $F(2, 23) = 229.117$, $p = .0001$, $\eta_p^2 = .952$]. The results of the post-hoc analysis revealed that the participants in the LC condition ($M = 22.72$, $SD = 5.86$) spent less time inside the collaboration zone than that during the MC condition ($M = 59.16$, $SD = 9.28$), at $p = .001$, and the HC condition ($M = 84.97$, $SD = 17.81$), at $p = .001$. Moreover, the participants in the MC condition spent less time inside the collaboration zones compared to that in the HC condition, at $p = .001$.

Table 3. Descriptive statistics of the in-game measurements across the three experimental conditions (LC: Low Collaboration, MC: Medium Collaboration, and HC: High Collaboration), and the obtained results.

Condition	*M*	*SD*	*Min*	*Max*	Results
			Degree of Collaboration		
LC	.17	.06	.05	.39	LC<MC (p = .001)
MC	.40	.03	.32	.47	MC<HC (p = .001)
HC	.45	.04	.38	.55	LC<HC (p = .0001)
			Player Distance (in cm)		
LC	111.21	67.54	55.87	384.46	no significant result
MC	102.11	16.73	75.04	140.92	
HC	111.45	12.75	79.93	133.96	
			Travel Distance (in cm)		
LC	642.69	36.90	585.54	770.46	LC<MC (p = .001)
MC	717.40	58.20	638.04	832.21	MC<HC (p = .007)
HC	799.19	93.41	611.40	969.68	LC<HC (p = .0001)
			Completion Time (in sec)		
LC	110.73	16.54	85.34	143.86	LC<MC (p = .001)
MC	146.15	24.61	90.46	191.09	MC<HC (p = .002)
HC	178.91	31.70	112.67	236.13	LC<HC (p = .001)
			Collaboration Time (in sec)		
LC	22.72	5.86	11.40	35.64	LC<MC (p = .001)
MC	59.16	9.28	41.24	77.64	MC<HC (p = .001)
HC	84.97	17.81	58.77	140.89	LC<HC (p = .001)
			Close Proximity Time (in sec)		
LC	4.30	4.11	.29	15.14	no significant result
MC	3.53	1.61	.41	8.22	
HC	4.30	1.45	.94	6.93	

5.6.2 Subjective Ratings. The perceived collaboration was also statistically significant across the examined conditions [$\Lambda = .469$, $F(2, 48) = 27.145$, $p = .0001$, $\eta_p^2 = .231$]. The results of the post-hoc analysis revealed that the participants rated the LC condition ($M = 4.93$, $SD = 1.80$) lower than the MC condition ($M = 6.31$, $SD = .91$), at $p = .001$, and the HC condition ($M = 6.54$, $SD = .72$), at $p = .001$. However, no statistically significant result was found between the MC and HC conditions ($p = .102$). Table 4 shows the descriptive statistics for the perceived collaborations.

5.6.3 Participant-Agent Correlation. We also explored how the participants collaborated during the gameplay compared to the AI virtual agents used to characterize the degree of collaboration of the developed game level chunk. For this part of the study, we isolated the per-game level chunk data collected from our participants. For the Pearson product-moment correlation analyses, we used the data obtained from the AI virtual agents for each

Table 4. Descriptive statistics of the perceived collaboration ratings across the three experimental conditions (LC: Low Collaboration, MC: Medium Collaboration, and HC: High Collaboration) and the obtained results.

Condition	***M***	***SD***	***Min***	***Max***	**Results**
Perceived Collaboration					
LC	4.93	1.80	1.17	7.00	LC<MC (p = .001)
MC	6.31	.91	3.34	7.00	LC<HC (p = .001)
HC	6.54	.72	4.00	7.00	

game level chunk and the averages obtained from the participants for each given game level chunk for all (15) game level chunks. Table 5 summarizes the raw numerical values used to compare the results obtained with the AI virtual agents and those obtained from our participants.

Table 5. Raw numerical values used to compare the results obtained with AI virtual agents (AI) and those obtained from our participants (P).

	Degree of Collaboration		Player Distance		Travel Distance		Completion Time		Collaboration Time		Close Proximity Time	
Chunk ID	P	AI	P	AI	P	AI	P	AI	P	AI	P	AI
C1	.02168	.21659	11.53515	3.84015	41.84316	37.36840	5.86087	10.05723	.00000	1.05340	.02168	.03501
C2	.07466	.21131	8.18353	5.13270	95.31888	109.64185	15.44542	31.55432	2.50024	5.65169	.07239	.00461
C3	.29778	.21744	12.42533	.58894	41.68053	40.08104	5.68682	10.00639	1.56606	2.25398	.02089	.08487
C4	.24246	.32782	9.58411	5.49595	46.83138	43.56703	7.31520	11.35065	1.78127	4.44627	.03168	.07620
C5	.30332	.27382	10.32784	2.27646	48.70996	36.08767	8.45367	9.45878	2.52987	2.78206	.03438	.04000
C6	.59434	.51531	4.60391	.62813	53.81859	40.15609	14.56040	10.04309	8.43210	5.45890	.02910	.06998
C7	.52461	.49580	11.64476	1.94619	54.15456	43.55383	10.66392	11.60944	5.55907	6.67865	.04141	.06698
C8	.69114	.52015	12.45848	12.33019	89.22649	81.91881	16.36952	26.34777	11.36309	12.64062	.03146	.00431
C9	.63797	.45949	9.62962	3.21142	68.36594	45.38921	14.48255	15.63601	9.25626	7.16166	.04872	.01937
C10	.65318	.70475	17.79996	14.54406	208.75490	106.59528	44.00969	77.62875	28.12076	46.55848	.05104	.04315
C11	.12335	.40382	11.88042	14.28864	82.99025	93.58184	21.11941	29.03274	2.56256	9.43818	.05524	.00000
C12	.09761	.43350	8.61864	10.38947	65.35683	68.06172	13.91274	25.77981	1.15355	8.10278	.03064	.01729
C13	.13913	.77391	16.44345	12.89808	93.89479	67.26303	26.08352	25.73684	3.65488	13.28221	.02949	.01305
C14	.78093	.71462	11.23880	12.58725	66.39209	41.04868	19.88709	10.83099	15.90273	5.63966	.02114	.00474
C15	.78348	.76937	12.34321	4.75234	68.78322	66.57927	15.00534	17.69756	11.71757	16.40833	.01260	.00399

The results of our analyses revealed a moderate positive correlation for the **degree of collaboration** variables (AI virtual agents and participants; $r = .604$, $n = 15$, $p = .004$), a moderate positive correlation for the **player distance** variables ($r = .613$, $n = 15$, $p = .012$), a strong positive correlation for the **travel distance** variables ($r = .811$, $n = 15$, $p = .0001$), a strong positive correlation for the **completion time** variables ($r = .896$, $n = 15$, $p = .0001$), and a strong positive correlation for the **collaboration time** variables ($r = .835$, $n = 15$, $p = .0001$). No significant correlation was observed for the **close proximity time** variables ($r = -.033$, $n = 15$, $p = .902$).

5.7 Discussion

We collected both objective data related to how the participants interacted in the synthesized game levels and subjective self-reported ratings to understand whether we could use our method to synthesize game levels that enforce a different collaboration gameplay behavior for our participants. The first glance at our results indicated that, although we used the degree of collaboration as the most important cost term of our total cost function (the assigned weight for the mean degree of collaboration cost was $w^M_{\text{Collab}} = 1.00$, while most other costs had weights < 1.00), four (**degree of collaboration**, **travel distance**, **completion time**, and **collaboration time**) out of the six measurements revealed a similar pattern: the measurements under the LC condition were lower than those under the MC and HC conditions, and the measurements under the MC condition were lower than those under the HC condition. Based on these findings, we argue that an optimization-based method can synthesize game levels that impact the collaboration gameplay behavior of our participants.

In terms of the **degree of collaboration** measurement, we observed an offset between the requested degree of collaboration targets (ρ_M = .30 for the LC, ρ_M = .50 for the MC, and ρ_M = .70 for the HC condition) and the actual collected data (.17 for the LC, .40 for the MC, and .45 for the HC condition) from our participants. The mean degree of collaboration of our participants was closer to the target degree of collaboration under the MC (.10 offset) and LC (.13 offset) conditions compared to the HC (.25 offset) condition. According to the literature [37, 41, 48], such an offset exists between the requested and actual values. In our method, the initial characterizations of the game level chunks from AI virtual agents were the main cause of such differences. We scripted the AI virtual agents to complete the task as efficiently as possible without being influenced by other parameters that might have impacted the participants (e.g., time of day, mood, and prior virtual reality and gameplay experiences). In addition, the participant groups were randomly composed, which meant that each participant also had to quickly understand the gameplay behavior of their partner during the study and build their gameplay strategy based upon that. Therefore, the main cause of the mentioned offsets could be the optimality of the AI virtual agents to execute and solve the given tasks.

Two of the examined measurements (**player distance** and **close proximity time**) were not significant. These findings indicate that the participants did not try to be in close proximity of each other; instead, each participant tried to build their own strategy during the gameplay. By combining both the significant and non-significant results, we realized that although the participants were planning their gameplay strategy independently, they planned it in such a way that would benefit the team and not only themselves, which is a typical behavior found in games [3, 18, 78]. Our findings indicated that our participants collaborated to progress the game by building their own strategies; therefore, a collaborative culture was maintained and built between the participants who worked together toward finishing the game.

Although we noted the offset between the requested degree of collaboration and the actual collected data, the correlation findings were also notable; they showed that the participants could perform their tasks in parallel with the AI virtual agents. According to the literature, AI virtual agents can be used to evaluate the difficulty of game levels [7, 54, 76, 85]. Our study extends such knowledge by revealing that AI virtual agents can also be used to evaluate the degree of collaboration that characterizes a game level; therefore, it extends the potential usage of AI virtual agents for evaluating not only the difficulty of a game level (as in [28, 55]) but also the degree of collaboration of game levels. However, as mentioned above, when game developers use AI virtual agents, they should always consider that such a method will return the optimal collaborative gameplay behavior and not the actual gameplay collaborative behavior that external or non-predefined parameters might influence.

Regarding the self-reported **perceived collaboration**, our participants perceived LC and HC as expected; however, they rated MC closer to HC. This result implies that the participants could not differentiate among the three conditions; however, the performed in-game measurements did not support this assumption. Either the targets for the degree of collaboration assigned to the mean degree of collaboration cost term were too close, or after a certain degree of collaboration, it was difficult for our participants to subjectively distinguish the degree of collaboration between the game levels (MC and HC conditions in our case). Another potential explanation for this finding could be how our participants interpreted each game level's "mean" collaboration target and how they reflected such interpretation on their understanding of the provided questions and their responses. For example, the participants might have thought more in terms of "max" degrees of collaboration for a given game level instead of the "mean" degree of that game level. Thus, instead of interpreting how much they collaborated by averaging their collaborative behavior across a whole level, they might have interpreted how much they collaborated in the game level chunk where they had to collaborate the most. According to the literature, individual cognitive styles impact collaborative gameplay [2, 85]. Moreover, by considering that increased self-esteem [83], self-efficacy [14], and self-motivation [25] can affect the perceived performance [11, 24] of participants, we should conduct further experimentation to properly understand and interpret how participants perceive different degrees of collaboration during gameplay.

Another cause that could have limited the results is that our method may not have linearly mapped spatial collaboration with the perceived collaboration of our participants. This could have been the case for two reasons. First, a spatial approach for defining collaboration between two entities could be considered somewhat limited, or its applicability could be restricted to only a small number of collaborative tasks. According to the Tang et al. [75] styles of coupling, it is obvious that people can be in the same area and work on different problems (the "different problems" style of coupling); therefore, a spatial measurement would not necessarily describe the collaboration between people. Second, another potential explanation is participants' potential overestimation of their relative contributions to collaborative endeavors [56], which means that capturing the perceived collaboration through self-reported data could also limit our understanding of how participants perceived their collaboration.

Furthermore, we collected comments from our participants to better understand their gaming experience regarding the three examined game levels (LC, MC, and HC game levels). Most participants indicated that they considerably enjoyed the collaborative experience in the gaming environment, and many said that they liked the game they played. One participant wrote, "*This was a great experience and a really enjoyable game. I definitely felt the collaborative atmosphere and felt that we worked well together.*" Another commented, "*I think that the easier the level, the less the players are inclined to collaborate with each other.*" One other participant wrote, "*The more complex puzzles made it much more necessary to interact with the other participant and made finishing them a lot more satisfying.*" Thus, according to the collected comments, the participants not only enjoyed the developed game levels but also understood that they had to build collaborative gameplay behavior with their partners.

Additionally, some participants noted the importance of communication in facilitating their collaboration. In particular, one wrote, "*I feel like my partner and I were always communicating about what we needed and were able to work well together.*" Another elaborated, "*During the simulation, my partner and I were able to communicate and collaborate to reach our end goal, which was to finish all the levels. We were able to develop plans to finish the levels successfully and within a decent amount of time. We were also able to finish the levels correctly.*" Note that, although we did not ask the participants to communicate during the gameplay, we observed that they were communicating. Based on our observations, as the target degree of collaboration of the game level increased, the communication between the participants also increased. This finding aligns with those of the previous studies conducted in the field [10, 12, 50, 77] that explored and analyzed the collaboration behavior of the participants during gameplay.

6 LIMITATIONS

Synthesizing game levels for collaborative gameplay is a complex process that requires numerous components to work harmoniously. Although the proposed pipeline can synthesize game levels for collaborative gameplay, we should also report the limitations. Note that these limitations do not invalidate our pipeline toward developing an automatic method for synthesizing game levels that satisfy the degrees of collaboration targets and other design decisions. Instead, they can help future research toward further advancement of the design of game levels for collaborative gameplay.

In this project, we demonstrated a simple approach to synthesize a game level, which we characterized as highly structured and linear. We think that conducting additional experiments in which we distribute collaboration-related tasks in an open-space virtual environment or form a non-linear method (e.g., similar to the work of Ma et al. [42]) of synthesizing game levels (e.g., having a game level chunk that may offer two branches to get through to a common destination) would help us further understand the collaborative gameplay behavior of the participants. In addition, we considered only two players collaborating to finish the game. However, in multiplayer games, we found more than two players; therefore, it is unclear how an increased number of players can affect our results.

The developed game level chunks that we used in our project impacted our project. In particular, the developed game level chunks were context-dependent and, thus, highly reliant on the designer's decisions. Given that game level and gameplay designers can use different approaches to enforce collaboration, it would be useful to develop guidelines to help researchers and developers more easily develop collaborative tasks for games. Furthermore, it remains unclear how our results would be affected when we use a larger number of game level chunks to compose a game level; this is something that we should certainly explore. Finally, you might have noticed, especially in Fig. 5, that some chunks (e.g., C15 in Fig. 5(f)) were repeated twice toward the end of the chunk sequence, but the line graph was strictly increasing. We think that developing a dataset with more than 15 game level chunks can introduce more variations in the degree of collaboration of the game level chunks so that our method can more closely match the targets requested by the game designer.

Many collaborative games (such as *Portal 2*[2]) and soccer games (such as *FIFA*[22]) require players to position themselves strategically across a sizable area rather than in close proximity, and other types of collaborations do not depend on any spatial relationship at all (similar to collaborations that occur in *Keep Talking and Nobody Explodes*[5]). Our method addresses only one particular aspect of player collaboration—a collaboration that requires physical proximity and task completion by two players—which we consider a limitation, given the potential variety of collaborative gameplay that game designers can develop.

In addition, we developed behavior trees to force our AI virtual agents to collaborate to finish each designed game level chunk to characterize the degree of collaboration of each game level chunk. The developed behavior trees were considered highly structured and did not allow the AI agents to explore potential alternatives. Moreover, the behavior trees did not contain actions such as "do nothing" or "do something not related to the given game level chunk." Such additional behaviors can help introduce even more variations in our trials during the automatic annotation process; however, it can also make the simulation run longer and might not capture the optimal collaborative behavior required to finish each game level chunk. In addition, instead of manually defining the collaboration zones, we can predict them using AI virtual agents; this is an additional direction we should further explore. Moreover, asking a few people playing the game level chunks can provide additional data that we can use besides the data provided by the AI virtual agents to augment the annotation of each game level chunk, thus complementing the automatic annotation pipelines. The abovementioned approach can lead to generalized and improved methods for characterizing the degree of collaboration at any game level. All these limitations should be further explored in future studies.

It will be interesting to collect data on the collaboration "in the real world," such as chatting. In our study, the participants were co-located in the same room; thus, collecting the data on the time they spent discussing their strategy could have provided additional measurements to evaluate their collaborative behavior. Moreover, we should have collected measurements to capture the interactions that each player contributed to finishing the provided game level, such as each player's actions toward task completion (e.g., button clicks and gestures). Finally, including additional questionnaires, such as a questionnaire on presence [63] and questions related to mutual awareness and dependent actions [9], could have helped us to understand the overall experiences of our participants.

Lastly, our current study does not encompass real-world collaboration or how virtual reality collaboration could be translated into real-world collaboration, which we consider an additional limitation. However, we think that such a method could be used for automatically synthesizing serious games, such as virtual reality skill training applications (e.g., fire evacuation training) [79], which benefit skills acquisition and retention [62]. In such a case, trainees could experience variations in training scenarios with different degrees of collaboration, which could potentially benefit their real-world collaboration.

[22]https://en.wikipedia.org/wiki/FIFA_(video_game_series)

7 CONCLUSIONS AND FUTURE WORK

We developed a method that considers the degree of collaboration the players are exposed to when playing a game. Our method provides game developers with the freedom to control various parameters of cost terms, allowing them to design game levels with specified objectives. To understand the potential of our method to synthesize game levels with different degrees of collaboration objectives, we conducted a user study and collected both in-game measurements and subjective ratings. We found that the degree of collaboration targets of the synthesized game level of our method impacted the way the participants collaborated in the gaming application.

In the future, we will work to synthesize collaboration-aware game levels for multiple players. We would also like to extend and evaluate our method to analyze less structured game levels. Moreover, we wish to explore the potential of using collaboration-aware games as a training tool to improve the collaborative behavior required by game players when playing games of various genres. Given that defining gameplay collaboration is an under-explored domain and that collaboration is task- and objective-dependent, we should conduct additional research toward developing a more generalized method for controlling the degree of collaboration required for different game levels and game genres. Finally, to further understand the collaborative gameplay behavior of the participants, we will conduct additional studies to compare collaboration behaviors in which people perform tasks such as those presented in this paper while being co-located in the same room with instructions to communicate and those not to communicate and being in separate rooms with chat functionality enabled. Such study conditions would help us further understand how the players perform the various tasks encoded in the game level chunks and how they communicate to coordinate in such tasks.

REFERENCES

[1] Paul S Adler and Seok-Woo Kwon. 2002. Social capital: Prospects for a new concept. *Academy of management review* 27, 1 (2002), 17–40.

[2] Sultan A Alharthi, George E Raptis, Christina Katsini, Igor Dolgov, Lennart E Nacke, and Z O Toups. 2021. Investigating the effects of individual cognitive styles on collaborative gameplay. *ACM Transactions on Computer-Human Interaction (TOCHI)* 28, 4 (2021), 1–49.

[3] Sultan A Alharthi, Ruth C Torres, Ahmed S Khalaf, Zachary O Toups, Igor Dolgov, and Lennart E Nacke. 2018. Investigating the impact of annotation interfaces on player performance in distributed multiplayer games. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–13.

[4] Alba Amato. 2017. Procedural content generation in the game industry. In *Game Dynamics*. Springer, New York City, New York, USA, 15–25.

[5] Aaron Bauer and Zoran Popović. 2012. RRT-based game level analysis, visualization, and visual refinement. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. AAAI, Palo Alto, California, 811–13.

[6] Glen Berseth, M Brandon Haworth, Mubbasir Kapadia, and Petros Faloutsos. 2014. Characterizing and optimizing game level difficulty. In *International Conference on Motion in Games*. ACM, New York, NY, USA, 153–160.

[7] Glen Berseth, M Brandon Haworth, Mubbasir Kapadia, and Petros Faloutsos. 2014. Characterizing and optimizing game level difficulty. In *Proceedings of the Seventh International Conference on Motion in Games*. ACM, New York, NY, USA, 153–160.

[8] Glen Berseth, Mubbasir Kapadia, and Petros Faloutsos. 2013. Steerplex: Estimating scenario complexity for simulated crowds. In *Proceedings of Motion on Games*. ACM, New York, NY, USA, 67–76.

[9] Frank Biocca, Chad Harms, and Jenn Gregg. 2001. The networked minds measure of social presence: Pilot test of the factor structure and concurrent validity. In *4th annual international workshop on presence, Philadelphia, PA*. PBWorks, San Mateo, CA, 1–9.

[10] Diego Buchinger and Marcelo da Silva Hounsell. 2018. Guidelines for designing and using collaborative-competitive serious games. *Computers & Education* 118 (2018), 133–149.

[11] Justin V Cavallo, John G Holmes, Gráinne M Fitzsimons, Sandra L Murray, and Joanne V Wood. 2012. Managing motivational conflict: How self-esteem and executive resources influence self-regulatory responses to risk. *Journal of personality and social psychology* 103, 3 (2012), 430.

[12] Victor Cheung, Y-L Betty Chang, and Stacey D Scott. 2012. Communication channels and awareness cues in collocated collaborative time-critical gaming. In *ACM Conference on Computer Supported Cooperative Work*. ACM, New York, NY, USA, 569–578.

[13] Siddhartha Chib and Edward Greenberg. 1995. Understanding the metropolis-hastings algorithm. *The american statistician* 49, 4 (1995), 327–335.

[14] Huey-Wen Chou. 2001. Influences of cognitive style and training method on training effectiveness. *Computers & Education* 37, 1 (2001), 11–25.

[15] Jacob Cohen. 2013. *Statistical power analysis for the behavioral sciences.* Academic press, Cambridge, MA.
[16] Michele Colledanchise and Petter Ögren. 2018. *Behavior trees in robotics and AI: An introduction.* CRC Press, Boca Raton, Florida.
[17] Siobhán Corrigan, GDR Zon, A Maij, Nick McDonald, and L Mårtensson. 2015. An approach to collaborative learning and the serious game development. *Cognition, Technology & Work* 17, 2 (2015), 269–278.
[18] Susan Danby, Ann-Carita Evaldsson, Helen Melander, and Pål Aarsand. 2018. Situated collaboration and problem solving in young children's digital gameplay. *British Journal of Educational Technology* 49, 5 (2018), 959–972.
[19] Fernando de Mesentier Silva, Scott Lee, Julian Togelius, and Andy Nealen. 2017. AI-based playtesting of contemporary board games. In *International Conference on the Foundations of Digital Games.* ACM, New York, NY, USA, 1–10.
[20] Ciro Donalek, S George Djorgovski, Alex Cioc, Anwell Wang, Jerry Zhang, Elizabeth Lawler, Stacy Yeh, Ashish Mahabal, Matthew Graham, Andrew Drake, et al. 2014. Immersive and collaborative data visualization using virtual reality platforms. In *IEEE International Conference on Big Data.* IEEE, New York, NY, USA, 609–614.
[21] Yanan Fan and Scott A Sisson. 2011. Reversible jump MCMC. Handbook of Markov Chain Monte Carlo.
[22] Franz Faul, Edgar Erdfelder, Axel Buchner, and Albert-Georg Lang. 2009. Statistical power analyses using G* Power 3.1: Tests for correlation and regression analyses. *Behavior research methods* 41, 4 (2009), 1149–1160.
[23] 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.
[24] Nigel Ford. 2000. Cognitive styles and virtual environments. *Journal of the American Society for information science* 51, 6 (2000), 543–557.
[25] Nigel Ford and Sherry Y Chen. 2001. Matching/mismatching revisited: An empirical study of learning and teaching styles. *British Journal of Educational Technology* 32, 1 (2001), 5–22.
[26] Anthony Francis. 2019. Overcoming Pitfalls in Behavior Tree Design. In *Game AI Pro 360: Guide to Architecture.* CRC Press, Boca Raton, Florida, 309–320.
[27] Linus Gisslén, Andy Eakins, Camilo Gordillo, Joakim Bergdahl, and Konrad Tollmar. 2021. Adversarial reinforcement learning for procedural content generation. In *IEEE Conference on Games.* IEEE, IEEE, New York City, New York, USA, 1–8.
[28] Miguel González-Duque, Rasmus Berg Palm, David Ha, and Sebastian Risi. 2020. Finding Game Levels with the Right Difficulty in a Few Trials through Intelligent Trial-and-Error. In *2020 IEEE Conference on Games (CoG).* IEEE, New York, NY, USA, 503–510.
[29] Camilo Gordillo, Joakim Bergdahl, Konrad Tollmar, and Linus Gisslén. 2021. Improving playtesting coverage via curiosity driven reinforcement learning agents. arXiv preprint arXiv:2103.13798.
[30] Peter J Green. 1995. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* 82, 4 (1995), 711–732.
[31] Scott W Greenwald, Wiley Corning, and Pattie Maes. 2017. Multi-user framework for collaboration and co-creation in virtual reality. In *International Conference on Computer Supported Collaborative Learning.* ACM, 1–2.
[32] Christian Güttler and Troels Degn Johansson. 2003. Spatial principles of level-design in multi-player first-person shooters. In *Proceedings of the 2nd workshop on Network and system support for games.* ACM, New York, NY, USA, 158–170.
[33] 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* 9, 1 (2013), 1–22.
[34] Hikaru Ibayashi, Yuta Sugiura, Daisuke Sakamoto, Natsuki Miyata, Mitsunori Tada, Takashi Okuma, Takeshi Kurata, Masaaki Mochimaru, and Takeo Igarashi. 2015. Dollhouse vr: a multi-view, multi-user collaborative design workspace with vr technology. In *SIGGRAPH Asia Emerging Technologies.* ACM, New York, NY, USA, 1–2.
[35] 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.
[36] A Donald Keedwell and József Dénes. 2015. *Latin squares and their applications.* Elsevier, Amsterdam, Netherlands.
[37] Wanwan Li, Biao Xie, Yongqi Zhang, Walter Meiss, Haikun Huang, and Lap-Fai Yu. 2020. Exertion-aware path generation. *ACM Transactions on Graphics* 39, 4 (2020), 115–1.
[38] 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.* AAAI, Palo Alto, California, 30–36.
[39] Can Liu, Olivier Chapuis, Michel Beaudouin-Lafon, and Eric Lecolinet. 2016. Shared interaction on a wall-sized display in a data manipulation task. In *ACM CHI Conference on Human Factors in Computing Systems.* ACM, New York, NY, USA, 2075–2086.
[40] 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.
[41] Huimin Liu, Zhiquan Wang, Christos Mousas, and Dominic Kao. 2020. Virtual Reality Racket Sports: Virtual Drills for Exercise and Training. In *IEEE International Symposium on Mixed and Augmented Reality.* IEEE, New York, NY, USA, 566–576.
[42] Chongyang Ma, Nicholas Vining, Sylvain Lefebvre, and Alla Sheffer. 2014. Game level layout from design specification. *Computer Graphics Forum* 33, 2 (2014), 95–104.
[43] Ali Ahmad Malik, Tariq Masood, and Arne Bilberg. 2020. Virtual reality in manufacturing: immersive and collaborative artificial-reality in design of human-robot workspace. *International Journal of Computer Integrated Manufacturing* 33, 1 (2020), 22–37.

[44] Alicia P Melis, Brian Hare, and Michael Tomasello. 2006. Chimpanzees recruit the best collaborators. *Science* 311, 5765 (2006), 1297–1300.

[45] Liang Men and Nick Bryan-Kinns. 2018. LeMo: supporting collaborative music making in virtual reality. In *IEEE VR workshop on sonic interactions for virtual environments*. IEEE, New York, NY, USA, 1–6.

[46] Kathryn E Merrick, Amitay Isaacs, Michael Barlow, and Ning Gu. 2013. A shape grammar approach to computational creativity and procedural content generation in massively multiplayer online role playing games. *Entertainment Computing* 4, 2 (2013), 115–130.

[47] Marçal Mora-Cantallops and Miguel-Ángel Sicilia. 2018. MOBA games: A literature review. *Entertainment computing* 26 (2018), 128–138.

[48] 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* 11, 2 (2021), 1–19.

[49] Thammathip Piumsomboon, Youngho Lee, Gun Lee, and Mark Billinghurst. 2017. CoVAR: a collaborative virtual and augmented reality system for remote collaboration. In *SIGGRAPH Asia Emerging Technologies*. ACM, New York, NY, USA, 1–2.

[50] Hayo Reinders and Sorada Wattana. 2014. Can I say something? The effects of digital game play on willingness to communicate. *Language Learning & Technology* 18, 2 (2014), 101–123.

[51] Christian Reuter. 2016. *Authoring Collaborative Multiplayer Games-Game Design Patterns, Structural Verification, Collaborative Balancing and Rapid Prototyping*. Ph. D. Dissertation. Technische Universität Darmstadt.

[52] Christian Reuter, Viktor Wendel, Stefan Göbel, and Ralf Steinmetz. 2014. Game Design Patterns for Collaborative Player Interactions. In *DiGRA*. DiGRA, 1–16.

[53] José Bernardo Rocha, Samuel Mascarenhas, and Rui Prada. 2008. Game mechanics for cooperative games. In *ZON Digital Games*. ZON, 72–80.

[54] Shaghayegh Roohi, Christian Guckelsberger, Asko Relas, Henri Heiskanen, Jari Takatalo, and Perttu Hämäläinen. 2021. Predicting Game Difficulty and Engagement Using AI Players. *Proceedings of the ACM on Human-Computer Interaction* 5, CHI PLAY (2021), 1–17.

[55] Shaghayegh Roohi, Asko Relas, Jari Takatalo, Henri Heiskanen, and Perttu Hämäläinen. 2020. Predicting Game Difficulty and Churn Without Players. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. ACM, New York, NY, USA, 585–593.

[56] Kenneth Savitsky, Leaf Van Boven, Nicholas Epley, and Wayne M Wight. 2005. The unpacking effect in allocations of responsibility for group tasks. *Journal of Experimental Social Psychology* 41, 5 (2005), 447–457.

[57] Jesse Schell. 2008. *The Art of Game Design: A book of lenses*. CRC press, New York, NY, USA.

[58] Carolina Islas Sedano, Maira B Carvalho, Nicola Secco, and C Shaun Longstreet. 2013. Collaborative and cooperative games: Facts and assumptions. In *2013 International Conference on Collaboration Technologies and Systems*. IEEE, New York, NY, USA, 370–376.

[59] Magy Seif El-Nasr, Bardia Aghabeigi, David Milam, Mona Erfani, Beth Lameman, Hamid Maygoli, and Sang Mah. 2010. Understanding and evaluating cooperative games. In *ACM SIGCHI conference on human factors in computing systems*. ACM, New York, NY, USA, 253–262.

[60] Yoones A Sekhavat. 2017. Behavior trees for computer games. *International Journal on Artificial Intelligence Tools* 26, 02 (2017), 1730001.

[61] Alexander Shoulson, Max L Gilbert, Mubbasir Kapadia, and Norman I Badler. 2013. An event-centric planning approach for dynamic real-time narrative. In *Motion on Games*. ACM, New York, NY, USA, 121–130.

[62] Ka-Chun Siu, Bradley J Best, Jong Wook Kim, Dmitry Oleynikov, and Frank E Ritter. 2016. Adaptive virtual reality training to optimize military medical skills acquisition and retention. *Military medicine* 181, suppl_5 (2016), 214–220.

[63] Mel Slater, Martin Usoh, and Anthony Steed. 1994. Depth of presence in virtual environments. *Presence: Teleoperators & Virtual Environments* 3, 2 (1994), 130–144.

[64] Adam M Smith and Michael Mateas. 2011. Answer set programming for procedural content generation: A design space approach. *IEEE Transactions on Computational Intelligence and AI in Games* 3, 3 (2011), 187–200.

[65] Robert Sugden. 2015. Team reasoning and intentional cooperation for mutual benefit. *Journal of Social Ontology* 1, 1 (2015), 143–166.

[66] Anthony Tang, Melanie Tory, Barry Po, Petra Neumann, and Sheelagh Carpendale. 2006. Collaborative coupling over tabletop displays. In *ACM SIGCHI conference on Human Factors in computing systems*. ACM, New York, NY, USA, 1181–1190.

[67] Ann Marie Thomson, James L Perry, and Theodore K Miller. 2009. Conceptualizing and measuring collaboration. *Journal of public administration research and theory* 19, 1 (2009), 23–56.

[68] Julian Togelius, Noor Shaker, and Mark J Nelson. 2014. *Procedural Content Generation in Games: A Textbook and an Overview of Current Research*. Springer, Berlin, Germany.

[69] 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.

[70] Wenpin Tsai and Sumantra Ghoshal. 1998. Social capital and value creation: The role of intrafirm networks. *Academy of management Journal* 41, 4 (1998), 464–476.

[71] April Tyack, Peta Wyeth, and Daniel Johnson. 2016. The appeal of moba games: What makes people start, stay, and stop. In *ACM Annual Symposium on Computer-Human Interaction in Play*. ACM, New York, NY, USA, 313–325.

[72] Lorraine M Uhlaner, Ilse A Matser, Marta M Berent-Braun, and Roberto H Flören. 2015. Linking bonding and bridging ownership social capital in private firms: Moderating effects of ownership–management overlap and family firm identity. *Family Business Review* 28, 3 (2015), 260–277.

[73] Benjamin van Arkel, Daniel Karavolos, Anders Bouwer, Sander Bakkes, and Frank Nack. 2015. Procedural generation of collaborative puzzle-platform game levels. In *International Conference on Intelligent Games and Simulation*. Eurosis, 87–93.
[74] Riemer van Rozen and Quinten Heijn. 2018. Measuring quality of grammars for procedural level generation. In *International Conference on the Foundations of Digital Games*. ACM, New York, NY, USA, 1–8.
[75] Tong Wang and Shuichi Kurabayashi. 2020. Sketch2map: a game map design support system allowing quick hand sketch prototyping. In *IEEE Conference on Games*. IEEE, New York, NY, USA, 596–599.
[76] Daniel Wheat, Martin Masek, Chiou Peng Lam, and Philip Hingston. 2015. Dynamic difficulty adjustment in 2d platformers through agent-based procedural level generation. In *2015 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE, New York, NY, USA, 2778–2785.
[77] Brian M Winn and JW Fisher. 2004. Design of communication, competition, and collaboration in online games. In *Dipresentasikan dalam computer game technology conference*. Citeseer, 1–13.
[78] Jason Wuertz, Sultan A Alharthi, William A Hamilton, Scott Bateman, Carl Gutwin, Anthony Tang, Zachary Toups, and Jessica Hammer. 2018. A design framework for awareness cues in distributed multiplayer games. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–14.
[79] Biao Xie, Huimin Liu, Rawan Alghofaili, Yongqi Zhang, Yeling Jiang, Flavio Destri Lobo, Changyang Li, Wanwan Li, Haikun Huang, Mesut Akdere, Christos Mousas, and Lap-Fai Yu. 2021. A review on virtual reality skill training applications. *Frontiers in Virtual Reality* 2 (2021), 645153.
[80] 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.
[81] José P Zagal, Jochen Rick, and Idris Hsi. 2006. Collaborative games: Lessons learned from board games. *Simulation & gaming* 37, 1 (2006), 24–40.
[82] Natalia Padilla Zea, José Luís González Sánchez, Francisco L Gutiérrez, Marcelino J Cabrera, and Patricia Paderewski. 2009. Design of educational multiplayer videogames: A vision from collaborative learning. *Advances in Engineering Software* 40, 12 (2009), 1251–1260.
[83] Li-fang Zhang. 2006. Does student–teacher thinking style match/mismatch matter in students' achievement? *Educational psychology* 26, 3 (2006), 395–409.
[84] Zhuoming Zhou, Elena Márquez Segura, Jared Duval, Michael John, and Katherine Isbister. 2019. Astaire: A Collaborative Mixed Reality Dance Game for Collocated Players. In *Annual Symposium on Computer-Human Interaction in Play*. ACM, New York, NY, USA, 5–18.
[85] Mohammad Zohaib. 2018. Dynamic difficulty adjustment (DDA) in computer games: A review. *Advances in Human-Computer Interaction* 2018 (2018), 1–13.

A APPENDIX: THE BEHAVIOR TREES

In this section, we present the developed behavior trees, which summarize the major events used in our game level chunks. Behavior trees describe switchings between a finite set of tasks in a modular fashion and control the execution flow of the tasks. Events can invoke other events during their execution. Please refer to previously published work on behavior trees [16, 26, 60] for a detailed description of the implementation process. Here, we provide a brief description of the main components of the behavior trees:

- **Composite:** A composite node is a node that can have one or more children. Such a node processes one or more of these children in either a first to last sequence or random order depending on the particular composite node in question. In addition, at some stage, it considers their processing complete and passes either success or failure to the parent, which is often determined by the success or failure of the child nodes. During the time a composite node is processing children, it continues to return "Running" to the parent.
- **Decorator (or Decor):** A decorator node, like a composite node, can have a child node. Unlike a composite node, a decorator node can only have a single child. The decorator node's function is either to transform the result it received from its child node's status to terminate the child, or to repeat processing of the child, depending on the type of decorator node.
- **Leaf:** Leafs are the most powerful node type, as they are defined and implemented to command the game-specific actions. An example of this, as used in the behavior trees implemented in this project, is "Go to the target." A "Go to the target" leaf node makes the AI virtual agent walk to a specific position in the game level chunk and return success or failure, depending on the result. Because we can define what leaf nodes are, they can be very expressive when layered on top of composite and decor nodes and

allow the developer to make powerful behavior trees capable of quite complicated layered and intelligently prioritized behaviors.

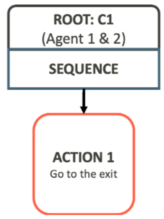


Fig. 7. Behavior tree for the C1 game level chunk (Nodes: 2; Depth: 1).

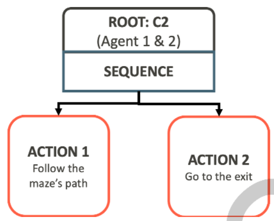


Fig. 8. Behavior tree for the C2 game level chunk (Nodes: 3; Depth: 1).

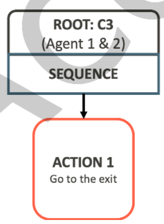


Fig. 9. Behavior tree for the C3 game level chunk (Nodes: 2; Depth: 1).

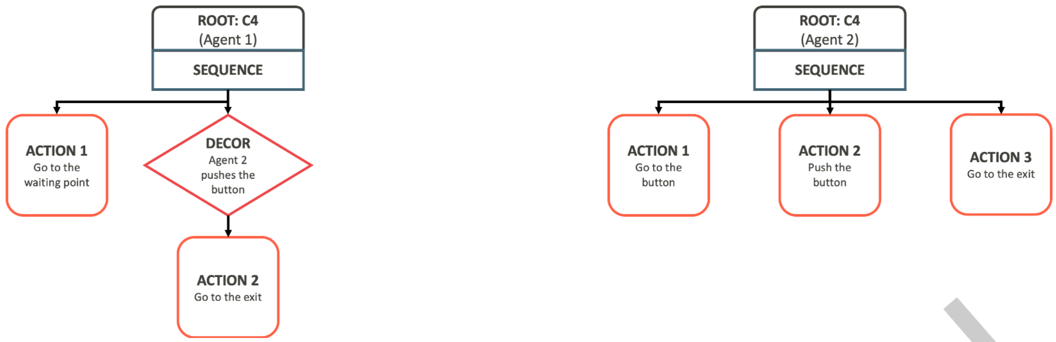


Fig. 10. Behavior trees for the C4 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 4; Depth: 1];).

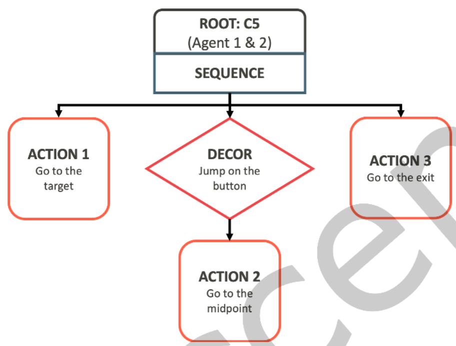


Fig. 11. Behavior tree for the C5 game level chunk (Nodes: 5; Depth: 2).

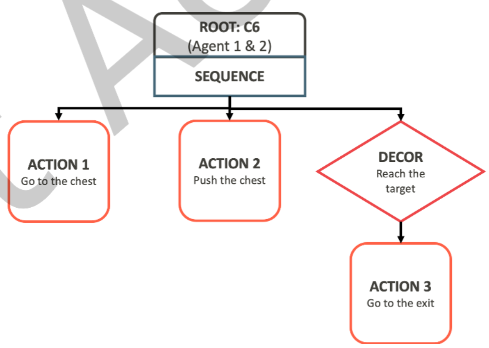


Fig. 12. Behavior tree for the C6 game level chunk (Nodes: 5; Depth: 2).

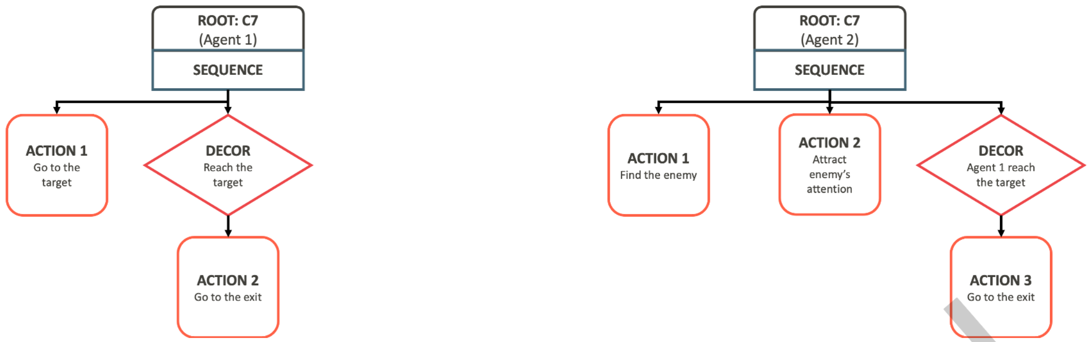


Fig. 13. Behavior trees for the C7 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2];).

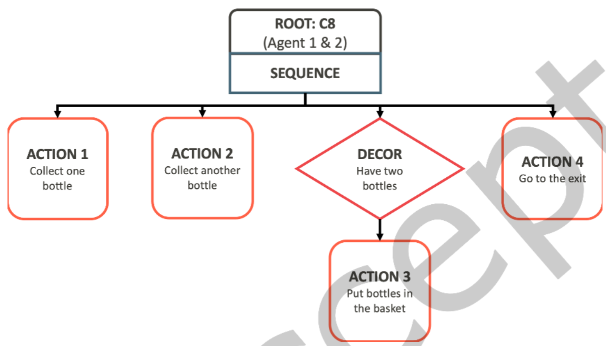


Fig. 14. Behavior tree for the C8 game level chunk (Nodes: 6; Depth: 2).

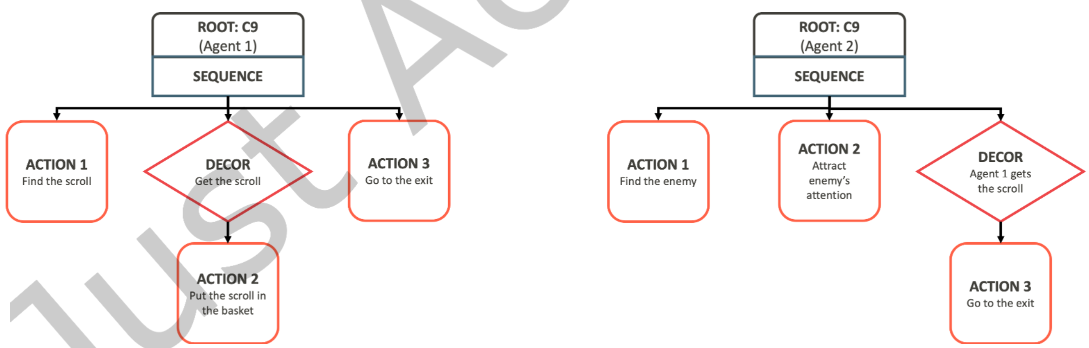


Fig. 15. Behavior trees for the C9 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2];).

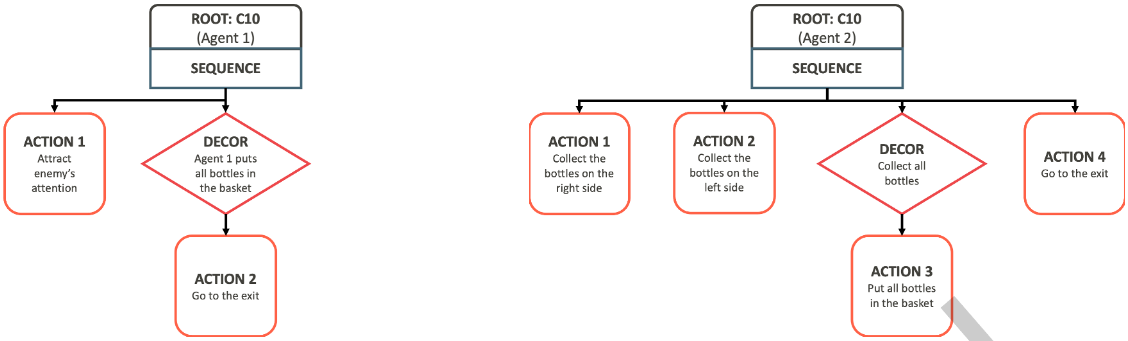


Fig. 16. Behavior trees for the C10 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 6; Depth: 2];).

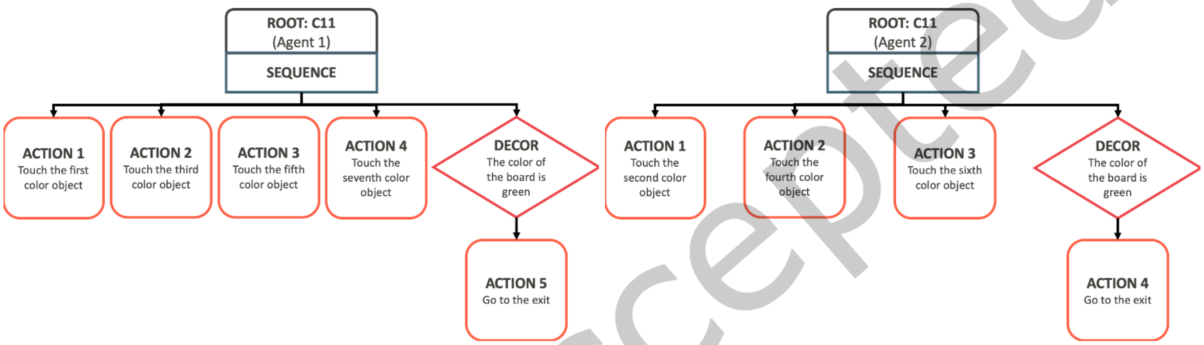


Fig. 17. Behavior trees for the C11 game level chunk (Left: Player 1 [Nodes: 7; Depth: 2]; Right: Player 2 [Nodes: 6; Depth: 2];).

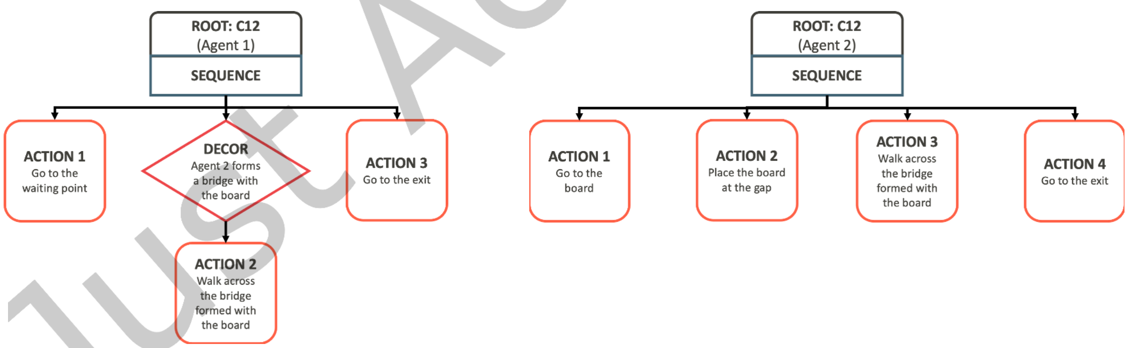


Fig. 18. Behavior trees for the C12 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 1];).

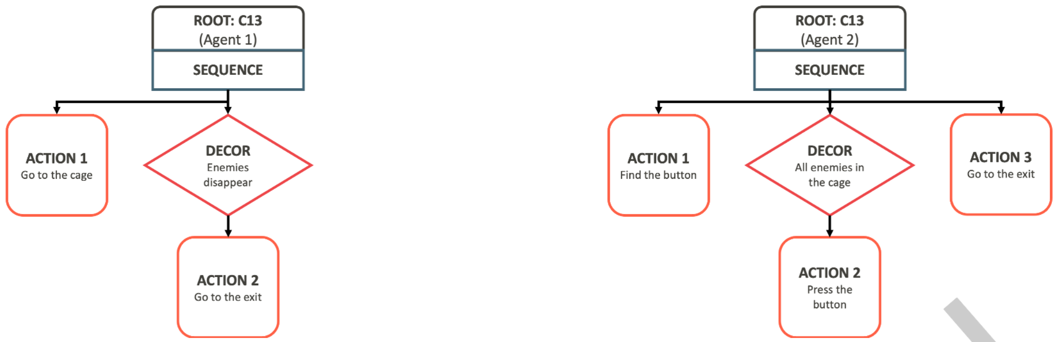


Fig. 19. Behavior trees for the C13 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2];).

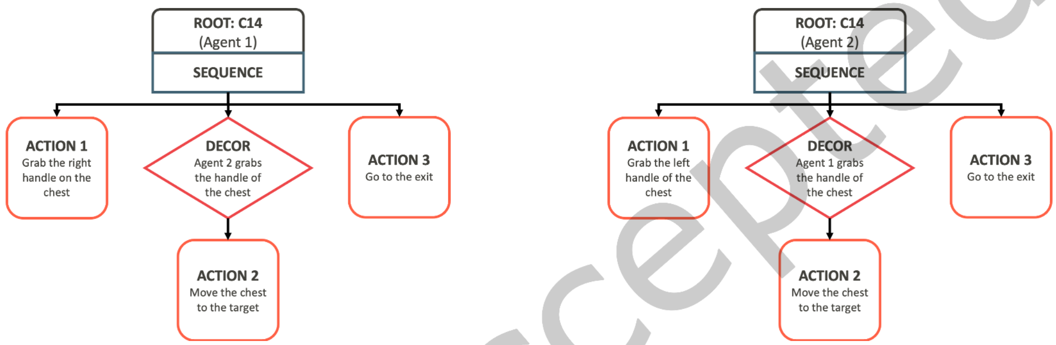


Fig. 20. Behavior trees for the C14 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2];).

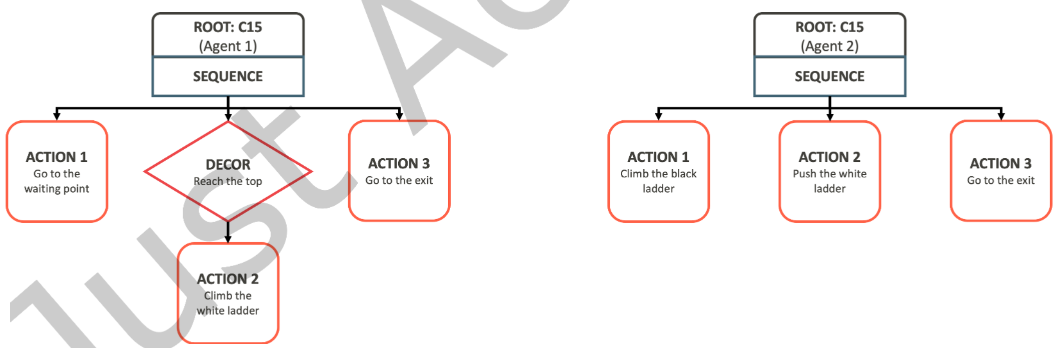


Fig. 21. Behavior trees for the C15 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 4; Depth: 1];).