

**SUPPORTING CONTENT GENERATION IN VIRTUAL
ENVIRONMENTS BASED ON OPTIMIZATION**

by
Huimin Liu

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy



Department of Computer Graphics Technology

West Lafayette, Indiana

May 2023

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Christos Mousas, Chair

Department of Computer Graphics Technology

Dr. Dominic Kao

Department of Computer and Information Technology

Dr. David Whittinghill

Department of Computer Graphics Technology

Dr. Nandhine Giri

Department of Computer Graphics Technology

Approved by:

Dr. Kathryn A. Newton

ACKNOWLEDGMENTS

Completing a Ph.D. was a challenge for me, and I would not have been able to do it without the invaluable assistance of my advisor, Dr. Christos Mousas, and the members of my committee. Throughout the entire process, they provided me with insights and feedback that proved to be essential for my academic success. I am incredibly grateful to Dr. Mousas for his kindness and generosity in guiding me through my Ph.D. study. I feel incredibly fortunate to have chosen him as my Ph.D. supervisor. Chris is not just an excellent teacher but also a good friend who always considers problems from the student's point of view, respects my wishes, and provides helpful guidance and constructive advice. He is always willing to answer my questions, even the seemingly "stupid" ones, with patience and clarity. I owe him so much, and I cannot express enough gratitude for his responsible guidance, which has not only helped me grow academically but also personally.

Moreover, I would like to thank Professors Dominic Kao, David Whittinghill, and Nandhini Giri for their unique insights and detailed feedback on my defense and dissertation. Their input allowed me to examine my Ph.D. project from multiple perspectives and broaden my horizons. Without their help, I could not have come this far.

I would also like to express my gratitude to Professor Craig Yu from George Mason University for his kind support and professional insights in my research work. His detailed proofreading of my paper manuscript was incredibly valuable, and I look forward to the chance to work with him and the students from CMU in the future.

I would like to acknowledge the significant contributions made by Zhiquan Wang, Angshuman Mazumda, Minsoo Choi, Pedro Acevedo, Liuchuan Yu, and Koilias Alexandros, who provided me with their kind help and support on all my projects. Without their assistance, I could not have achieved so much.

Finally, I want to express my sincere appreciation to all the students at Purdue VR Lab. We worked together for four years, facing deadlines together, celebrating achievements together, and dining with our advisor and everyone got along harmoniously. Even though I was in a foreign country far from home, I never felt lonely because of their friendship.

I hope that they all have a bright future ahead of them, as they deserve all the success that they will undoubtedly achieve.

TABLE OF CONTENTS

LIST OF TABLES	8
LIST OF FIGURES	9
ABSTRACT.....	13
INTRODUCTION	14
CHAPTER 1. PUBLISHED ARTICLE #1: VIRTUAL REALITY RACKET SPORTS: VIRTUAL DRILLS FOR EXERCISE AND TRAINING	25
1.1 Introduction.....	26
1.2 Related Work	28
1.3 Synthesizing Racket Sports Exercise Drills.....	29
1.3.1 Shot Terms Cost	31
1.3.1.1 Distance Cost.....	31
1.3.1.2 Speed Cost.....	32
1.3.1.3 Frequency Cost.....	32
1.3.2 Prior Cost	32
1.3.2.1 Duration Cost	33
1.3.2.2 Variations Cost	33
1.3.2.3 Court Side Cost	33
1.3.3 Optimization	34
1.4 Evaluation as An Exercise Tool.....	37
1.4.1 Participants	37
1.4.2 Setup and Implementation Details.....	37
1.4.3 Conditions of the Study	38
1.4.4 Measurements	39
1.4.5 Procedure	39
1.4.6 Results.....	40
1.4.7 Discussion.....	41
1.5 Evaluation as A Training Tool.....	43
1.5.1 Participants	43
1.5.2 Study Details.....	44

1.5.3	Results.....	46
1.5.4	Discussion.....	48
1.6	Limitations	49
1.7	Conclusions and Future Work Directions.....	50
1.8	References.....	50
CHAPTER 2. PUBLISHED ARTICLE #2: VIRTUAL REALITY GAME LEVEL LAYOUT DESIGN FOR REAL ENVIRONMENT CONSTRAINTS.....		59
2.1	Introduction.....	60
2.2	Related Work	62
2.3	Preliminaries	64
2.3.1	Capturing the real environment and Constraints	64
2.3.2	Game level chunks.....	65
2.3.3	Grid generation and characterization.....	66
2.4	Problem formulation and optimization	67
2.4.1	Mapping cost	68
2.4.2	Fitting cost	69
2.4.3	Variations cost	69
2.4.4	Accessibility cost.....	70
2.4.5	Optimization	71
2.5	Implementation details and example games	73
2.5.1	Implementation	73
2.5.2	Example games.....	75
2.6	User study	76
2.6.1	Participants	77
2.6.2	Experimental conditions	77
2.6.3	Measurements	78
2.6.4	Procedure	79
2.6.5	Results.....	79
2.6.6	Discussion.....	81
2.7	Conclusions, limitations, and future work	83
2.8	Declaration of Competing Interest.....	84

2.9	References	84
CHAPTER 3. PUBLISHED ARTICLE #3: SYNTHESIZING GAME LEVELS FOR COLLABORATIVE GAMEPLAY IN A SHARED VIRTUAL ENVIRONMENT.....		
		89
3.1	Introduction.....	90
3.2	Related Work	93
3.3	Preliminary Remarks	99
3.3.1	Game Level Chunks	101
3.3.2	Game Level Chunk Characterization.....	103
3.4	Problem Formulation and Optimization	105
3.4.1	Formulation.....	106
3.4.2	Collaboration Costs	106
3.4.3	Prior Costs	110
3.4.4	Optimization	110
3.5	User Study.....	113
3.5.1	Participants	114
3.5.2	Setup and Implementation Details.....	114
3.5.3	Experimental Conditions	115
3.5.4	Measurements	117
3.5.5	Procedure	118
3.5.6	Results.....	119
3.5.6.1	In-game Measurements.	119
3.5.6.2	Subjective Ratings.....	121
3.5.6.3	Participant-Agent Correlation.	122
3.5.7	Discussion.....	124
3.6	Limitations	127
3.7	Conclusions and Future Work	129
3.8	References.....	130
CHAPTER 4. PUBLISHED ARTICLE #4: SYNTHESIZING SHARED SPACE VIRTUAL REALITY FIRE EVACUATION TRAINING DRILLS		
		138
4.1	Introduction.....	139
4.2	Related Work	141

4.3	Preliminary Remarks	142
4.3.1	System Overview	142
4.3.2	Environment Representation	142
4.3.3	Virtual Training Environment	142
4.3.4	Authoring Training Drill.....	144
4.4	Problem Formulation	144
4.5	Optimization	146
4.6	User Study.....	147
4.6.1	Participants	148
4.6.2	Conditions.....	148
4.6.3	Measurements	149
4.6.4	Procedure	149
4.6.5	Setup and Implementation Details.....	150
4.6.6	Results.....	151
4.6.6.1	Objective Data.....	151
4.6.6.2	Subjective Self-reported Data	151
4.6.7	Discussion.....	152
4.6.8	Limitations	153
4.7	Conclusion	154
4.8	Acknowledgements.....	154
4.9	References.....	154
	DISCUSSION AND CONCLUSION	158
	APPENDIX.....	159
	PUBLISHED ARTICLES	167

LIST OF TABLES

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 lev 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).	105
Table 2: Perceived Collaboration Scale used in this study.	118
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.....	121
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.....	122
Table 5: Raw numerical values used to compare the results obtained with AI virtual agents (AI) and those obtained from our participants (P).	123

LIST OF FIGURES

Figure 1: Our approach can optimize virtual reality exercise and training drills for different racket sports with minimal effort from the user. From left to right: table tennis, badminton, and mini tennis. © [2020] IEEE.....	27
Figure 2: Total cost changes as the optimization process (iterations) evolves. © [2020] IEEE...	35
Figure 3: Variations in exercise drills generated by the presented system through keeping the target exercise amounts ($\sigma D = .6$, $\sigma V = .3$, and $\sigma \Phi = .2$) constant. Numbers close to balls denote the sequence of shots. Note that for all four examples shown in this Figure, the weights of the shot and prior cost terms remained constant. © [2020] IEEE	36
Figure 4: Distribution of shots when varying the target values (σD , σV , and $\sigma \Phi$) of the shots cost terms. Numbers close to balls denote the sequence of shots. Note that for all four examples shown in this figure the weight of the shot and prior cost terms remained constant. © [2020] IEEE	36
Figure 5: The virtual table tennis court that was designed and used for the purpose of the study. © [2020] IEEE	38
Figure 6: Boxplots of the obtained results regarding the usage of the developed method as an exercise tool. Boxes enclose the middle 50% of the data. The median is denoted by a thick horizontal line. © [2020] IEEE	41
Figure 7: Comparison across the three examined conditions (NT: no training, VRT: virtual reality training, and RWT: real-world training) used to evaluate the developed virtual reality table tennis application. Negative and low values indicate that our participants' performance did not improve at the post-training session. Positive values indicate that participant performance improved. Boxes enclose the middle 50% of the data. The median is denoted by a thick horizontal line. © [2020] IEEE.....	47
Figure 8: Given a real environment and its constraints, the user can easily capture its layout using a virtual reality controller. Later, our proposed method synthesizes a game level layout that matches the real environment layout and its constraints. © [2021] Elsevier.....	61
Figure 9: Examples of game level chunks developed for the proposed project. © [2021] Elsevier	66
Figure 10: A layout of the real environment (left) and its representation based in the virtual grid (right). W stands for walkable, O stands for obstacle, and B stands for boundary grid cells. © [2021] Elsevier	67
Figure 11: The layout of an intermediate game level synthesized using the Perlin noise (a), and the layout of the final synthesized game level layout in which the Perlin noise is included with different σV targets (4(b)–(e)). For all examples, the variation cost is given high priority, $wV = 1.00$, and the accessibility cost is given low priority, $wA = .05$. © [2021] Elsevier	70

Figure 12: Example of synthesized level layouts based on different input layouts and weights assigned to each cost term of the total cost function. For all examples, the weight of the variation cost is set to $wV = .00$. © [2021] Elsevier	73
Figure 13: Number of iterations needed by our system to synthesize game level layouts with different grid resolutions (5×5 , 10×10 , 15×15 , and 20×20 grids) and different percentages (0%, 25%, 50%, 75% and 100%) of obstacles that occupy the grids. © [2021] Elsevier	74
Figure 14: A comparison between optimizing the game level layout using the MCMC and the greedy algorithm in a 5×5 grid. The MCMC algorithm achieves lower minima compared to the greedy algorithm. © [2021] Elsevier	74
Figure 15: Screenshots of the Backyard Fortune (left) and The Rebooter (right) games. © [2021] Elsevier	76
Figure 16: A user playing our Backyard Fortune game in our lab space. © [2021] Elsevier.....	76
Figure 17: Boxplots from self-reported ratings for each examined concept across the three experimental conditions. © [2021] Elsevier	81
Figure 18: Our method synthesizes a game level in which participants collaborate in a shared virtual environment to play a game.	93
Figure 19: 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.	100
Figure 20: Example scenes of the developed game level chunks from a first-person perspective.	103
Figure 21: 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 $wCollabM = 1.00$, $wCollabV = .30$, and $wCollabP = .50$, and those of the prior cost terms at $wPriorS = 1.00$ and $wPriorR = .50$. The same game level chunk can appear more than once at a synthesized level (e.g., C1, C3, and C5 in Figure 21(a)); however, due to the adjacent repetition cost term, the system does not repeat the same chunk one after the other.....	108
Figure 22: Example game levels ($\rho S = 9$) using different degrees of collaboration progress line graphs while maintaining the mean degree of collaboration target constant. For all examples, we use $\rho M = .50$ and $\rho V = .50$ as the targets.	113
Figure 23: 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.	117
Figure 24: Two players in different locations, wearing a VR headset on the VR treadmill. Their task is to guide the agents out of the building where a simulated fire emergency occurs. The two players are in the same virtual space even though their physical locations are different. They can communicate, use voice commands to guide the agents outside the building, and use a fire	

extinguisher to eliminate the fire in the building. We illustrate users' and agents' positions and the top view of the building in the minimap. © [2022] IEEE	140
Figure 25: Different parts of the designed virtual environment we used in our prototype application. © [2022] IEEE	143
Figure 26: A user commands a virtual agent to “come here” and the agent moves toward the user. © [2022] IEEE	143
Figure 27: Example scenes from the synthesized training drill. © [2022] IEEE.....	148
Figure 28: The three experimental conditions we used for our user study. Top: The position and size of fires (orange cells), the optimal paths (blue cells), and the position of the virtual agents. Bottom: The visibility of each training drill. © [2022] IEEE.....	149
Figure 29: (a) Users can enable a minimap. The minimap provided information on players' position, the position of the virtual agents, the exits, and the commands they could use. (b) Two users collaborate in the shared virtual environment to open a path to escape the building. © [2022] IEEE	150
Figure 30: Behavior tree for the C1 game level chunk (Nodes: 2; Depth: 1).....	160
Figure 31: Behavior tree for the C2 game level chunk (Nodes: 3; Depth: 1).....	160
Figure 32: Behavior tree for the C3 game level chunk (Nodes: 2; Depth: 1).....	160
Figure 33: Behavior trees for the C4 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 4; Depth: 1] ;).....	161
Figure 34: Behavior tree for the C5 game level chunk (Nodes: 5; Depth: 2).....	161
Figure 35: Behavior tree for the C6 game level chunk (Nodes: 5; Depth: 2).....	162
Figure 36: Behavior trees for the C7 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2] ;).....	162
Figure 37: Behavior tree for the C8 game level chunk (Nodes: 6; Depth: 2).....	163
Figure 38: Behavior trees for the C9 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2] ;).....	163
Figure 39: Behavior trees for the C10 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 6; Depth: 2] ;).....	164
Figure 40: Behavior trees for the C11 game level chunk (Left: Player 1 [Nodes: 7; Depth: 2]; Right: Player 2 [Nodes: 6; Depth: 2] ;).....	164
Figure 41: Behavior trees for the C12 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 1] ;).....	165
Figure 42: Behavior trees for the C13 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2] ;).....	165

Figure 43: Behavior trees for the C14 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2] ;)	166
Figure 44: Behavior trees for the C15 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 4; Depth: 1] ;)	166

ABSTRACT

Content generation in virtual environments is becoming increasingly important with the rise of virtual and augmented reality technologies and growing demand for immersive experiences. This raises a problem of efficient content generation to meet with higher requirements for both the quantity and quality of the contents that could be used inside virtual environments. This dissertation explored the possibilities of formulating design problems as computational problems based on optimization theory in different scenarios, and explored what can be viable application cases, as well as what can be viable cost terms for each application case based on the theory. In total four application scenarios are included. The optimization theory used in this dissertation is the Markov chain Monte Carlo optimization method called “simulated annealing”. By doing this we can transform a design problem to a computation problem and use computational methods to quickly solve the problem and generate content.

This dissertation contains the papers published by the author during her Ph.D. Each published article included in this dissertation deals with a specific application case based on optimization theory.

The author investigated four distinct application cases. The first case centered on the synthesis of drills for virtual reality racket sports. The second case focused on designing virtual reality game level layouts, based on the layout of a real-world environment. The third case explored collaborative gameplay design, with the aim of synthesizing game levels that require a predetermined degree of collaboration between two players to complete. The aim of the fourth application case was to create virtual reality fire evacuation training drills that could be used for training purposes in simulated environments. Different cost terms are proposed based on different application cases to synthesize contents that align with the design intent.

Keywords: Generative Design, Virtual Reality, Procedural Content Generation, Optimization Techniques

INTRODUCTION

Introduction of Dissertation Research

Extended reality technologies develop at a high speed. Multiple research reports have shown investors have a highly optimistic attitude about the market of extended reality technologies. For example, the worldwide virtual reality market size was valued at USD 35.0 billion in the year 2023, with an anticipated compound annual growth rate (CAGR) of 13.8% expected from 2023 to 2030 [1]. The global augmented reality market was expected to grow at a compound annual growth rate (CAGR) of 40.9% from 2022 to 2030 and reach a market size of USD 597.54 billion by 2030 [2]. The emergence of new technologies such as virtual reality (VR) and augmented reality (AR) has had a significant impact on traditional content creation pipelines. These technologies offer new possibilities for creating immersive and interactive experiences, but they also present new challenges for content creators.

At the 2005 Game Developers Conference, Will Wright, a renowned game designer known for creating games such as Sim City, discussed the challenges that game developers face when creating content [3]. He referred to this challenge as the "Mountain of Content Problem," which refers to the difficulty in creating enough content within a limited amount of time while also keeping costs under control. Game developers face pressure to produce a sufficient amount of content that is engaging, enjoyable, and challenging for players, while at the same time also maintaining a reasonable price for their target audience. Moreover, with the rapidly evolving hardware devices and software tools, game developers must continuously adapt and learn new technologies to provide their customers with the best experience possible. However, the traditional content creation process is usually tedious, time-consuming, and labor-intensive, requires lots of manual work, the process of modification and adjustment is also complicated. According to data from VENNGAGE¹ blog [4], 36.7% of the participants reported one of the primary challenges faced by marketers is consistently creating engaging visual content. Marketers were asked to rate the difficulty level of producing visual content continuously on a scale of 1-10, 76.6% of marketers rated five or higher.

¹ <https://venngage.com/>

One of the existing research branches that focus on dealing with this problem is procedural content generation (PCG) [5]. Refers to methods that can procedurally generate virtual contents by computer programs. The content includes but is not limited to maps, levels, terrains, plants, gameplay, and game objects such as rocks, enemies, traps, etc. [6]. A recent research study indicates that the integration of PCG has the potential to enhance city-building games experience without substantially compromising players' ability to express their creativity [7]. Another similar research branch that has been gaining significant attention from society beyond the computer science community is AI-generated content (AIGC) [8], content generation products such as ChatGPT [9] and DALL-E [10] demonstrated their potential to revolutionize the way of content creation.

One of the most widely used approaches among PCG methods is optimization [11]. The general formulation process of optimization is to first formalize a mathematic model, also referred to as an objective function, taking several design aspects into consideration, encode those design considerations into the objective function, called cost terms. Then, a design problem can be transformed into a computational design problem which can be solved using mathematical methods [11].

This dissertation explored the possibilities of formulating design problems as computational problems based on optimization theory in different scenarios in virtual environments, and explored what can be viable application cases, as well as what can be viable cost terms for each application case based on the theory. In total four application scenarios are included. Each application case is described in detail in a published paper included as one chapter in this dissertation. The simulated annealing optimization algorithm based on Markov Chain Monte Carlo (MCMC) was applied as a common method for all the explored scenarios and was used to automatically synthesize contents in virtual environments.

Simulated annealing is a widely used random search optimization algorithm developed by the inspiration of metal thermal processing technology, first proposed in 1953 [12], and then was applied to combinatorial optimization [13]. The algorithm has been widely used in engineering to solve nondeterministic polynomial (NP) time complexity problems and to overcome limitations of local minimum in optimization process and initial value dependence. The idea is derived from the annealing process of heating a solid to a temperature high enough so that the molecules are randomly arranged, then gradually cooling them down, and finally the molecules are arranged in

a low-energy state to reach a stable state. More details about the algorithm theory and implementation please refer to [14]. While genetic algorithm is a random search algorithm based on the theory of natural selection and heredity, which combines the survival of the fittest in the process of biological evolution with the random exchange mechanism of chromosomes in the population [15]. Many basic concepts like evolution, locus and allele are derived from Charles Darwin's theory of evolution.

For more details regarding the MCMC simulated annealing and how the framework was applied in each explored scenario, please refer to the published articles included in Chapter 1, 2, 3 and 4.

Overview of Purpose

The purpose of this dissertation was to explore ways to formulate computational design problems based on the theory of MCMC optimization, to explore the possibilities of formulating design problems into a generalized optimization-based framework which could be solved easily using computational methods to help generate virtual contents. The aim was to explore ways to simplify the content creation process. The framework could encode designer's design considerations inside as cost terms then transform the design problem to computational problem which can be solved mathematically. The definition of the cost terms was different from scenario to scenario, and all that designer needs to consider was how to combine specific scenario domain knowledge to define cost terms to successfully generate the objective scenario.

Four application scenarios were explored in this dissertation, each one was supported by a published article of the author which was included as individual chapter of the dissertation in Chapter 1, 2, 3, and 4.

Significance of Research

Formulating design problem into computational design problems based on optimization make sense in following aspects:

1. It can help simplify the design process by automating certain aspects of design and reducing the time and effort required to manually search for optimal solutions, reducing the budget for content creation companies [16].

2. The design space can be more systematically explored to generate a large number of design alternatives, which can lead to better quality solutions.
3. Optimization-based computation design approaches offer a systematic and efficient framework for addressing complex design problems that involve numerous criteria or constraints.
4. It allows the designer to control the generated results according to her/his intents with minimal effort. This flexibility can increase the reusability of the generated contents and provide space for creativity.

Research Questions

Research questions addressed overall

This dissertation focused on generalizing scenario design problems based on optimization to support content generation in virtual environments. The author's previously published papers were included, each showed a different scenario design case. All the cases were based on the framework of MCMC optimization called "simulated annealing".

The research questions in this dissertation were summarized as below:

RQ1. Is it possible to formulate design problems based on optimization theory?

RQ2. What can be suitable/viable application cases based on the theory?

RQ3. What can be suitable/viable cost terms for each application case based on the theory?

All the design considerations were encoded as cost terms. A brief introduction of each design scenario and the discussion about the included cost terms were included below. For more details about the optimization theory and explanation for cost terms, please refer to the published papers included.

Research questions addressed in each published article

As described above, each published article demonstrated an application scenario in which a design problem was successfully formulated as a computational design problem based on optimization theory. The resulting synthesis was successful, producing different synthesized results based on different target cost input values and weights according to the theory, which is a

positive answer to RQ1. The synthesized result could trigger statistically significant differences in human behavior. This allowed designers to have a certain degree of control over the synthesized result, which served as a demonstration of the validity of the formulation and answered RQ2. Different customized cost terms related to specific domain knowledge based on the explored four particular scenarios were explored and discussed in detail in each article, which answered RQ3.

Application Case1 --- Virtual Reality Racket Sports: Virtual Drills for Exercise and Training

In this case, the design of virtual reality racket sports drills was formulated as an optimization problem. The goal was to synthesize drills for racket sports such as table tennis, tennis, badminton, and so on. The domain knowledge applied in this scenario was the factors or parameters that could affect the training/exercise intensity of the synthesized drill. By defining cost terms that were related to the gameplay and mechanics of the game and allowing user to control the parameters of the cost terms, user could easily adjust the objectives and intensity levels of the exercise drills. The synthesized results could be used for the purpose of training or exercise, and the effectiveness of the method was demonstrated by two studies. The first study investigated the potential usefulness of the developed virtual reality gaming application as an exercise tool by comparing its workout effectiveness at three intensity levels (low, medium, and high) through the collection of heart rate readings. The second study explored the potential utility of the virtual reality gaming application as a training tool by exploring whether there was any improvement in participants' performance across the three conditions (no training, virtual reality training, and real-world training). The results indicate that a virtual reality gaming application, such as the examined virtual reality table tennis exergame, could indeed be used effectively as both an exercise and a training tool. For more details, please refer to [17], which is also included in the Chapter 1 Published Article #1: Virtual Reality Racket Sports: Virtual Drills for Exercise and Training in this dissertation.

Discussion of how the paper address the research questions

First, the successfully synthesized result with certain degree of controllability to allow designer to change the synthesized result by changing the inputs of target values for cost terms and weights was a positive answer to RQ1 in the design of virtual reality racket sport scenario. Then, the resulting changes on human behavior were demonstrated as significant in user study

experiments, which was a valid proof showing the virtual reality racket sport design scenario was a suitable case to be formulated as optimization problem and could be solved procedurally, which answered RQ2.

The cost terms that contain specific domain knowledge considered in this case included *shot term cost* and *prior cost*. The *shot term cost* included considerations for *distance*, *speed*, and *frequency* for the generated shots, intended at controlling the training/exercise intensity of the synthesized drill. While the *prior cost terms* were developed to control some of the features of the gameplay before the user started to play. Various *prior cost terms* could have been employed, depending on the specific design requirements of an exercise. However, in this case, with racket sports domain knowledge applied, the prior cost terms that were explored were *duration*, *variation*, and *court side*. The exploration of the cost terms answered RQ3.

Application Case2 --- Virtual Reality Game Level Layout Design for Real Environment Constraints

In this case, the virtual reality environment's design was formulated as an optimization problem. The aim was to explore possible ways to integrate reality information, such as physical spatial constraints, into the optimization-based computational design framework to generate a virtual environment layout that aligned with the reality environment layout. It enabled realistic interaction with virtual environments as well as enhanced safety considerations for the generated results. The domain knowledge applied in this case was the physical environment layout information. Users first calibrated the environment layout manually using the Oculus Quest device set, and then the real environment's information was used as input for the target values for the cost terms in the formulated optimization problem. To evaluate the proposed method, a user study was conducted. The results indicated that the proposed method enhanced the levels of presence and involvement of participants in the virtual environment, and reduced the fear of collision with the real environment and its constraints. For more details, please refer to [18], which is included in Chapter 2 Published Article #2: Virtual Reality Game Level Layout Design for Real Environment Constraints" in this dissertation.

Discussion of how the paper address the research questions

Firstly, the accomplished synthesized virtual environment that integrated reality environment considerations was a positive answer to RQ1 in virtual environment design scenario. Secondly, the user study was conducted under *Optimization*, *No Optimization* and *No Obstacles* conditions. In the *Optimization* condition, the layout of the real environment and its obstacles were captured and used to automatically generate a game level layout with virtual obstacles in the exact positions of the real obstacles. In the *No Optimization* condition, the real environment obstacles were moved to different positions to create a mismatch between the real and virtual environments. The *No Obstacles* condition was a baseline condition where there are no obstacles present in the real environment, and the game level layout was the same as in the other conditions, with boundary game level chunks and obstacle chunks in their initial positions. The result of the user study showed participants did experience higher level of presence and involvement and experience less collisions during the process. This provided an answer to RQ2 as it showed the virtual reality environment design could be a suitable case to be formulated into a computational optimization design problem.

The virtual environment was represented as an assembly of chunks, the layout design decisions considered as cost terms included *mapping cost*, *fitting cost*, *variations cost*, and *accessibility cost* in a total cost function. The exploration for the cost terms answered RQ3 in this scenario.

Application Case3 --- Synthesizing Game Levels for Collaborative Gameplay In a shared virtual environment

In this case, the collaborative gameplay design was formulated as an optimization problem. In the case of designing collaborative gameplays in games and VR applications, the tasks requiring users to collaborate, and the degree of collaboration required to accomplish a given task are usually manually built or programmed by the game's designers, which is a tedious and time-consuming process. However, "Collaboration" is an elusive term with various definitions, designing for a collaboration task is usually an iterative process with a lot of experience-and-test manual adjustment resulting in low efficiency. The goal in this case was to overcome this issue. The proposed pipeline could automatically characterize the degree of collaboration of game level chunks and synthesize game levels with designer-defined degrees of collaboration targets.

AI calibrated domain knowledge regarding collaboration was first explored based on context information. Since there is no common definition for collaboration and the definition for collaboration usually depends on specific scenarios, fifteen collaborative game levels were designed at a preliminary stage. Then, the collaboration zone for each level was specified manually by the research team. The idea was adopted from [21], in which various patterns that enforce collaboration between players were described. Next, the collaboration degree for each game level was calibrated by pre-programmed behavior tree driven AI agents. Then, the calibrated collaboration degree for each pre-designed collaborative game level was used as domain knowledge to formulate the collaborative gameplay design problem into an optimization problem. 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. For more details, please refer to [19], which was included in Chapter 3 Published Article #3: Synthesizing Game Levels for Collaborative Gameplay In a shared virtual environment in this dissertation.

Discussion of how the paper address the research questions

The proposed method was divided into three parts. First, a game level designer was responsible for designing playable game level chunks. Second, artificial intelligence (AI) virtual agents were implemented to play the game level chunks. Data was collected from these agents and was used to characterize the degree of collaboration of each game level chunk. Third, by developing cost terms that encode various design decisions, the method can automatically synthesize a game level that fulfills all designer-specified design decisions.

To begin with, the achieved synthesized collaborative gameplay that integrated context-dependent collaboration degree considerations was a positive answer to RQ1. The formulation of the collaborative gameplay design problem to an optimization problem allowed the proposed system to synthesize several variations of game levels that satisfy the designer-defined parameters in a few seconds. Next, user study was conducted under synthesized *Low collaboration degree*, *Medium collaboration degree* and *High collaboration degree* collaborative gameplay levels. The result showed that the different degree of collaboration targets of the synthesized game level impacted the way the participants collaborated in the gaming application, which demonstrated that it is suitable and viable to formulate the collaborative gameplay design problem to an optimization

problem as an answer to RQ2. Then, the cost terms considered included *collaboration costs* and *prior costs*. The *collaboration costs* included considerations for *Mean Degree of Collaboration*, *Variation in the Degree of Collaboration*, and *Degree of Collaboration Progress*, aiming at providing certain level of controllability to the collaboration related features of the synthesized result. The *prior costs* that were explored were synthesized chunk total number *size cost* as well as *adjacent repetition cost*. The exploration of the cost terms answers RQ3.

Application Case4 --- Synthesizing Shared Space Virtual Reality Fire Evacuation Training Drills

In this case, the focused scenario was the fire evacuation training drill, which was formulated to be an optimization problem like the three cases described above. The aim was to synthesize VR fire evacuation training drills in a shared virtual space to explore the participants' collaboration behavior. The proposed optimization-based method can be used to automatically generate fire evacuation training drills with varying levels of difficulty. The users' assigned task was to help virtual agents evacuate the building as quickly as possible using predefined interaction mechanisms (voice commands, trigger fire extinguisher, physical locomotion, etc.). The participants can join the training drill from different locations and collaborate and communicate in a shared virtual space to accomplish the task. The proposed VR training drill authoring method was evaluated by a user study conducted among three synthesized training drills with different difficulty levels: *low difficulty (LD)*, *medium difficulty (MD)*, and *high difficulty (HD)*. Both in-game measurements and subjective ratings were collected to explore how the participants collaborate in such a VR setup. For more details, please refer to [20], which was also included in Chapter 4 Published Article #4: Synthesizing Shared Space Virtual Reality Fire Evacuation Training Drills in this dissertation.

Discussion of how the paper address the research questions

First, the attained synthesized result of fire evacuation training drills of varying level of difficulty was a positive answer to RQ1. Then, user studies were conducted to evaluate difference of participants' collaboration behavior under synthesized drills with different level of difficulty, and the results showed that the degree of collaboration targets of the synthesized game level impacted the way the participants collaborated in the gaming application. This showed that it is

viable to formulate the design of fire evacuation training drill into an optimization problem as an answer to RQ2. Finally, the domain knowledge applied in this case was the parameters that affect the difficulty level of the generated fire evacuation training drill. The cost terms that are considered here include *Length cost*, *Turn cost*, *Fire cost* and *Visibility cost*. The exploration for the cost terms answers RQ3.

References

- [1] Grand View Research. 2023. *Virtual Reality (VR) Market Size, Share & Trends Analysis Report By Component (Hardware, Software, Services), By Device (Head-Mounted Display, Gesture Tracking Device), By Application, By Region, And Segment Forecasts, 2023 - 2030* (Report ID: GVR-1-68038-831-2). Grand View Research. San Francisco, CA, USA
- [2] Research and Markets. 2022. *Augmented Reality Market Size, Share & Trends Analysis Report by Component (Hardware, Software), by Display (Head-Mounted Display & Smart Glass, Head-Up Display, Handheld Devices), by Application, by Region, and Segment Forecasts, 2022-2030* (Report ID: 4312493). Research and Markets. Dublin, Ireland.
- [3] Wikipedia Contributors. 2022. Will Wright (game designer). *Wikipedia, The Free Encyclopedia*. Available: [https://en.wikipedia.org/wiki/Will_Wright_\(game_designer\)](https://en.wikipedia.org/wiki/Will_Wright_(game_designer)).
- [4] Khoja, Nadya. 2022. 16 Visual Content Marketing Statistics to Know for 2022 [Infographic]. *Vennngage Blog*. <https://venngage.com/blog/visual-content-marketing-statistics/>
- [5] Shaker, Noor., Togelius, Julian. and Nelson, Mark. J. 2016. *Procedural content generation in games*. Springer.
- [6] Togelius, Julian., Yannakakis, Georgios. N., Stanley, Kenneth. O. and Browne, Cameron. 2011. Search-based procedural content generation: A taxonomy and survey. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3), 172-186.
- [7] Volkmar, Georg., Alexandrovsky, Dmitry., Eilks, Asmus. E., Queck, Dirk., Herrlich, Marc. and Malaka, Rainer. 2022. Effects of PCG on Creativity in Playful City-Building Environments in VR. *Proceedings of the ACM on Human-Computer Interaction*, 6(CHI PLAY), 1-20.
- [8] Wu, Jiayang., Gan, Wensheng., Chen, Zefeng., Wan, Shicheng. and Lin, Hong. 2023. AI-generated content (AIGC): A survey. *arXiv preprint arXiv:2304.06632*.
- [9] OpenAI, 2021. GPT-4, *OpenAI*. Available: <https://openai.com/research/gpt-4>

- [10] Ramesh, Aditya., Dhariwal, Prafulla., Nichol, Alex., Chu, Casey. and Chen, Mark. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.
- [11] Arora, Jasbir. S. 1990. Computational design optimization: a review and future directions. *Structural Safety*, 7(2-4), 131-148.
- [12] Metropolis, Nicholas., Rosenbluth, Arianna. W., Rosenbluth, Marshall. N., Teller, Augusta. H. and Teller, Edward. 1953. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6), 1087-1092.
- [13] Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P. 1983. Optimization by simulated annealing. *science*, 220(4598), 671-680.
- [14] Koziel, Slawomir. and Yang, Xin-She. (Eds.). 2011. *Computational optimization, methods and algorithms*. (Vol. 356). Springer.
- [15] Vose, Michael. D. 1999. *The simple genetic algorithm: foundations and theory*. MIT press.
- [16] Krish, Sivam. 2011. A practical generative design method. *Computer-Aided Design*, 43(1), 88-100.
- [17] Liu, Huimin., Wang, Zhiquan., Mousas, Christos. and Kao, Dominic. 2020. Virtual reality racket sports: Virtual drills for exercise and training. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (pp. 566-576). IEEE.
- [18] Liu, Huimin., Wang, Zhiquan., Mazumdar, Angshuman. and Mousas, C. 2021. Virtual reality game level layout design for real environment constraints. *Graphics and Visual Computing*, 4, 200020.
- [19] Liu, Huimin., Choi, Minsoo., Kao, Dominic. and Mousas, Christos. 2022. Synthesizing Game Levels for Collaborative Gameplay in a Shared Virtual Environment. *ACM Transactions on Interactive Intelligent Systems (TiiS)*.
- [20] Liu, Huimin., Choi, Minsoo., Yu, Liuchuan., Koiliias, Alexandros., Yu, Lap-Fai. and Mousas, Christos. 2022. Synthesizing Shared Space Virtual Reality Fire Evacuation Training Drills. In *2022 IEEE International Symposium on Mixed and Augmented Reality (Adjunct) (ISMAR Adjunct)* (pp. 459-464). IEEE.
- [21] Reuter Christian, Wendel Viktor, Göbel Stefan, and Steinmetz Ralf. 2014. Game design patterns for collaborative player interactions. In *DiGRA*. DiGRA, 1–16.

CHAPTER 1. PUBLISHED ARTICLE #1: VIRTUAL REALITY RACKET SPORTS: VIRTUAL DRILLS FOR EXERCISE AND TRAINING

All the authors in the paper agreed on the author of the dissertation to use this publication in her Ph.D. dissertation.

This paper was published and presented at *International Symposium on Mixed and Augmented Reality (ISMAR) 2020*.

© [2020] IEEE. Reprinted, with permission, from [Liu, Huimin., Wang, Zhiquan., Mousas, Christos. and Kao, Dominic, Virtual reality racket sports: Virtual drills for exercise and training. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2020, November]

(Cite) Liu, H., Wang, Z., Mousas, C., & Kao, D. (2020, November). Virtual reality racket sports: Virtual drills for exercise and training. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (pp. 566-576). IEEE. DOI: [10.1109/ISMAR50242.2020.00084](https://doi.org/10.1109/ISMAR50242.2020.00084)

The authors confirm contribution to the paper as follows:

- Study Conceptualization and Design: Mousas, Christos, Kao, Dominic;
- Methodology and/or Simulation Design: Mousas, Christos;
- Execution/Implementation: Liu, Huimin, Wang, Zhiquan;
- Data collection: Liu, Huimin;
- Analysis and interpretation of results: Mousas, Christos, and Liu, Huimin;
- Findings, Conclusions, and/or Recommendations: Mousas, Christos, Kao, Dominic;
- Draft manuscript preparation: Mousas, Christos, Kao, Dominic, Liu, Huimin.

All authors reviewed the results and approved the final version of the manuscript.
All authors in this paper agree on the usage of the paper in this dissertation.
Purdue does not require any formal permission.

Abstract

We have developed a modular virtual reality gaming application that can be used to synthesize exercise drills for racket sports. By defining cost terms that are related to the gameplay and the mechanics of the game, as well as by allowing a user to control the parameters of the cost terms, users can easily adjust the objectives and the intensity levels of the exercise drills. Based on the user-defined exercise objectives, a Markov chain Monte Carlo optimization method called “simulated annealing” was used to optimize the exercise drill. The effectiveness of the developed virtual reality gaming application was measured in two studies by using virtual reality table tennis as the evaluation tool. The first study investigated the potential usefulness of the developed virtual reality gaming application as an exercise tool by comparing its workout effectiveness at three intensity levels (low, medium, and high) through the collection of heart rate readings. The second study explored the potential utility of the virtual reality gaming application as a training tool by exploring whether there was any improvement in participants’ performance across the three conditions (no training, virtual reality training, and real-world training). The results indicate that a virtual reality gaming application, such as the examined virtual reality table tennis exergame, could indeed be used effectively as both an exercise and a training tool. Limitations and future research directions are discussed further below.

1.1 Introduction

Virtual reality has proven to be an excellent tool not only for entertainment purposes, but also for several other applications such as training [26][28][37][52], rehabilitation [33][61][83], human behavior exploration [41][60][63], and visualization [1][20]. The use of virtual reality in these domains allows the user to observe and interact with the provided content in a highly immersive environment while also being entertained [31][54]. With the widespread popularity of virtual reality devices and peripheral equipment, several real-world experiences can be converted into virtual ones and brought into one’s own living room. The use of virtual reality for exercise purposes can even have real-world benefits for some users. Specifically, by playing and simultaneously exercising, users can improve their physical health and fitness while being entertained [6][18][65].

Since quite a few people are interested in racket sports-related games² (e.g., table tennis, tennis, badminton, etc.), we decided to develop a modular virtual reality gaming application that can be used for exercise and training purposes by racket sports enthusiasts, focusing mainly on table tennis, badminton, and mini tennis. While the potential of virtual reality gaming applications for exercise and training is appealing, designing exercise drills may become tedious for the user, as the parameters have all been pre-set by the developer. Bearing this in mind, the virtual reality gaming application presented in this paper allows users to customize exercise drills, as our system is able to automatically optimize an exercise based on user-specified objectives.

The developed application was inspired by previous research on procedural content generation for exergames [45][86][87] and virtual reality applications related to racket sports [10][55][57][75]. Our approach took into account several parameters related to racket sports games and represented these parameters as cost terms to a total cost function. Next, an optimization-based approach was used to synthesize the racket sport drill. By formulating the design of the racket sport drills as an optimization problem, in a few seconds, several exercise drills could be generated by our system which is designed to maintain a balance among different design schemes while ensuring the necessary variability between different generated drills. This variability is important for keeping the user engaged. As shown in Figure 1 and the accompanying video, our approach can be applied to different types of racket sports.



Figure 1: Our approach can optimize virtual reality exercise and training drills for different racket sports with minimal effort from the user. From left to right: table tennis, badminton, and mini tennis. © [2020] IEEE

The focus of the paper is twofold: (1) develop an algorithm for automatically synthesizing exercise and training drills for racket sports, and (2) evaluate the impact of the synthesized exercise and training drills on human performance. The effectiveness of the developed application and the ability of our algorithm to efficiently synthesize exercise and training drills had to be evaluated,

² <https://www.worldatlas.com/articles/what-are-the-most-popular-sportsin-the-world.html>

so two user studies were conducted to determine our method's potential for use as an exercise and training tool. The results indicate that this type of virtual reality application can indeed be used for both exercise and training purposes. However, aside from the advantages of exercising in virtual reality, there are also some limitations that should be taken into account by the research community, something that may spur the development of additional advanced virtual reality interfaces applicable to exercising and training in virtual reality racket sports.

The remainder of this paper is organized as follows. Related works are presented in Section 1.2. The methodology and implementation details are presented in Section 1.3. The first user study and results are presented in Section 1.4, and the second user study and results are presented in Section 1.5. Various limitations are presented in Section 1.6. Finally, the conclusions and potential for future research are addressed in Section 1.7.

1.2 Related Work

Because traditional video games are generally associated with reduced energy expenditure on part of the players due to decreased physical activity [43], strategies that allow players to entertain themselves while also increasing physical activity have also been explored [27][35][50]. In response to the difficulty of developing effective strategies to promote physical activity [69], a category of games called exergaming [80][84], or active video games [8], has been developed to incorporate virtual reality technologies into video games. Generally, exergames allow players to perform various exercise activities from the comfort of their living rooms. Such games require physical output as a means of interaction and engagement with the game. Aside from the capacity of such games to be used for exercise and fun, exergames are also considered a credible alternative to conventional training. This has made it possible for exergames to be used in sports training [13][36], breathing training for increasing lung capacity [67], balance enhancement [44], weight control [82], and motor training [77].

The idea of using exergames to improve the health of users has been increasingly promoted by the research, development, and health/medical communities [68][73]. When comparing non-exergames with exergames, studies have indicated that the latter increase user enjoyment and intrinsic motivation levels [4][5][22][58][78]. So far, studies have validated the positive health effects of exergames [18][46] on weight loss in adolescents and adults [6][82], and on improved balance- and movement-related physical performance in the elderly [65][85]. Moreover, it has

been found that physical activity has positive effects on cognitive and also physical functions [15][16][25][53][64][76]. A notable example of the above is the collaboration between West Virginia high schools and the KONAMI gaming company, through which the arcade dance-based video game “Dance Dance Revolution”³ was included in the high school curriculum as a way to tackle youth obesity.⁴

When developing exercise games, an important factor a developer needs to take into account is the degree of physical exercise that is required by the user [66], since, according to a prior study, players derive more enjoyment from games that are neither too difficult nor too easy [81]. Though it is important to define physical exercise goals, when developing commercial games, customarily it is the developers who manually set these goals [23][32]. Thus, a challenge arises for developers to design an exercise gaming experience that can be used efficiently by users of varying ages and fitness levels. Fitts’ law [48] and precision of difficulty [49] can be employed so that exercise parameters can be controlled by users. In the current implementation, we considered several parameters related to racket sports and ultimately provided users with control over the output of their exercise drills. To automate the exercise or training drill synthesis process, procedural techniques can be efficiently applied [14]. Such procedural techniques allow the development of fast and scalable designs while variations across design outputs are also ensured. Note that such techniques have already been successfully implemented in various games [14][29][34][79][80]. Our developed procedural exercise drill design method was inspired by previous research and by recent approaches to automatic game-level synthesis for exercising [45][86][87]. Our application extends the current list of such exergames by proposing the use of racket sports, and evaluates the virtual reality table tennis exergame for its potential as an exercise and training tool.

1.3 Synthesizing Racket Sports Exercise Drills

A method was developed to synthesize exercise drills for virtual reality racket sports with respect to several factors defined as cost terms in a total cost function. Let $E = [s_1, s_2, \dots, s_N]$ denote an exercise drill, which consists of a number of $s_i \in E$ shots generated by our system (it is worth mentioning here that a virtual ball-throwing machine was used to generate the shots from

³ https://www.konami.com/games/asia/en/products/ddr_a/

⁴ <https://www.sfgate.com/business/article/Video-dance-game-to-be-used-in-schools-West-2542902.php>

the exact same position) and assembled in a sequential order, where s_i corresponds to any possible shot. The exercise drill E is designed by a total cost function $C_{Total}(E)$:

$$C_{Total}(E) = w_S^T C_S + w_P^T C_P \quad (1)$$

where $C_S = [C_S^{Dist}, C_S^{Speed}, C_S^{Freq}]$ is a vector of shot cost, and $w_S = [w_S^{Dist}, w_S^{Speed}, w_S^{Freq}]$ are weights that correspond to the cost terms. The C_S^{Dist} , C_S^{Speed} and C_S^{Freq} terms encode the intensity of the exercise drill: C_S^{Dist} denotes the distance covered by the user to complete the drill and is expressed as the distance between two adjacent shots (the distance is computed between the position $P(s_i)$ and $P(s_{i+1})$ of the adjacent shot s_i and s_{i+1} , respectively), C_S^{Speed} denotes the speed of the shots, and C_S^{Freq} denotes the frequency with which the shots are generated. Note that each shot s_i is represented by a target position $P(s_i)$, speed $V(s_i)$, and frequency $\Phi(s_i)$.

The prior cost term $C_P = [C_P^{Dur}, C_P^{Var}, C_P^{Side}]$ includes the prior costs associated with the developed exergame, such as the duration of the exercise drill (C_P^{Dur}), the variations between the shots (C_P^{Var}) and the court side (C_P^{Side}), and the $w_P = [w_P^{Dur}, w_P^{Var}, w_P^{Side}]$ are weights assigned to the prior cost terms. It should be noted that aside from the proposed cost terms, various other cost terms can be examined by the developers, depending on the characteristics of the exercise drill. For the cost terms, we employed a Gaussian model in order to evaluate the distance between the given objective and the target objective of the exercise drill. The source code (Unity3D project) of our racket sports application is available at our GitHub repository: <https://github.com/Hearurt/VR-TableTennis-System>.

All cost terms presented in the below sections were computed by using normalized values that lie within the minimum and the maximum range of each individual target. In finding the targets, eight non-athlete healthy students (four males and four females aged 19-23) were required to exercise for 60 minutes by playing multiple variations of the exergame at varying exercise intensities. During that time, combinations of various target values for each cost term were tested, and the heart rate (beats per minute) of each participant was recorded by using a heart rate sensor, the Polar OH1⁵. Based on this initial data collection, we were able to define the range and the target values of the individual cost terms. Finally, it should be noted that the target objective of the

⁵ <https://www.polar.com/us-en/products/accessories/oh1-optical-heartrate-sensor>

optimization process was the manipulation of exercise intensity, which in our case will be later evaluated (see Section 4) by collecting heart rate data and self-reported perceived intensity rating.

Note that although a number of methods could be used to generate exercise and training drills, we choose to implement an optimization-based method to solve the exercise and training drill synthesis problem. For example, rule-based methods often fail to select appropriate parameters for the desired outcome (especially when multiple parameters should be fulfilled simultaneously) and, in most cases, synthesize the output in a product-appropriate manner [11]. However, optimization technique iterates through hundreds of systematic draws from the model parameter space to search for solutions that fit all constraints set by a user, no matter how complex the problem is, which makes it fairly reliable [71] and easy to implement new constraints/cost terms. Moreover, optimization techniques allow the estimation of complex solutions in a fast and scalable fashion, which rule-based techniques fail to do.

1.3.1 Shot Terms Cost

The three shot terms responsible for generating a new exercise drill E are defined in this section.

1.3.1.1 Distance Cost

In various exercise drills, user movement within a space is quite common and, according to sports science, locomotive movement while exercising presents various benefits [19][74]. In order to calculate how much a user moves, it is assumed that there is a linear relationship between the distance of two adjacent shots (the distance of the positions of two balls the time point they bounce on the side of the user) and the distance that the user would need to cover when exercising. Thus, we defined a cost to compute the distance between the positions of two adjacent shots as:

$$C_S^{Dist}(E) = 1 - \exp\left(-\frac{\left(\frac{1}{|E|-1}\sum_{(s_i, s_{i+1})} D(P(s_i), P(s_{i+1})) - \sigma_D\right)^2}{2\sigma_D^2}\right) \quad (2)$$

where σ_D is the target distance covered between two adjacent shots, $D(P(s_i), P(s_{i+1}))$ denotes the distance between the positions $P(s_i)$ and $P(s_{i+1})$ of the two adjacent shots s_i and s_{i+1} , respectively, and $|E|$ denotes the total number of shots.

1.3.1.2 Speed Cost

According to sports science literature [7][21][56][72], the speed in which a ball moves in racket sports enhances the intensity of the exercise, as the athlete needs to be prepared to quickly decide and adjust his/her movement toward the direction of a moving ball. Thus, we included a cost term to compute the speed intensity involved in the exercise drill:

$$C_S^{Speed}(E) = 1 - \exp\left(-\frac{\left(\frac{1}{|E|}\sum s_i V(s_i) - \sigma_V\right)^2}{2\sigma_V^2}\right) \quad (3)$$

where σ_V is the target average ball speed in completing an exercise drill E , and $V(s_i)$ denotes the speed of the s_i shot.

1.3.1.3 Frequency Cost

The last term applied in our shot cost term is related to the frequency in which a new ball should be generated by the virtual ball-throwing machine. Based on various sources, we found that frequency is important in exercising, since high frequencies tend to keep an athlete vigilant as there is no time to rest between adjacent shots, resulting in a more intense workout [2][3][30][40][42][47]. Thus, a frequency cost term was developed to compute the frequency intensity involved in the exercise drills:

$$C_S^{Freq}(E) = 1 - \exp\left(-\frac{\left(\frac{1}{|E|}\sum s_i \Phi(s_i) - \sigma_\Phi\right)^2}{2\sigma_\Phi^2}\right) \quad (4)$$

where σ_Φ is the target average of the ball-throwing frequency in completing an exercise drill E , and $\Phi(s_i)$ denotes the frequency of the s_i shot.

1.3.2 Prior Cost

In this implementation, the prior cost terms were developed to control some of the features of game play. Various prior cost terms could have been employed, depending on the specific design requirements of an exercise. However, we limited the prior cost terms to those that are most important for this particular virtual reality gaming application: duration, variation, and court side.

1.3.2.1 Duration Cost

The duration cost is responsible to softly constrain the exercise drill to be of a certain duration, and it is defined as:

$$C_P^{Dur}(E) = 1 - \exp\left(-\frac{(\sum_{s_i} \tau(s_i) - \sigma_\tau)^2}{2\sigma_\tau^2}\right) \quad (5)$$

where $\tau(s_i)$ denotes the duration of a single shot and σ_τ denotes the target duration of the exercise drill.

1.3.2.2 Variations Cost

To keep the user engaged with the exercise drill—since an exercise without variation would become less interesting—a variation term was also implemented as an additional prior cost. When we perform exercise drills that require multiple repetitions of the same shot, the variation between repetitions should be minimized. Thus, the variation cost term ensures that the generated shots will or will not have the same characteristics, and it is defined as:

$$C_P^{Var}(E) = \frac{1}{|E|-1} \sum_{(s_i, s_{i+1})} \Gamma(s_i, s_{i+1}) \quad (6)$$

where (s_i, s_{i+1}) represents adjacent shots and $\Gamma(s_i, s_{i+1})$ returns 1 if the position and speed of shot s_{i+1} is identical to shot s_i (i.e., within a defined speed and position range); otherwise (s_i, s_{i+1}) returns 0 (the position and speed of shot s_{i+1} is different from shot s_i).

1.3.2.3 Court Side Cost

The court side cost is responsible for assigning a side to the synthesized drill, and it is defined as:

$$C_P^{Side}(E) = \frac{1}{|E|} \sum_{s_i} \Pi(s_i) \quad (7)$$

where $\Pi(s_i)$ returns 0 if the shot is generated at the chosen court side; otherwise $\Pi(s_i)$ returns 1. This cost term can be considered beneficial especially in cases where a racket sports enthusiast is willing to put in extra effort for a particular shot (e.g., forehand, backhand).

1.3.3 Optimization

An optimization approach was used to synthesize an exercise drill by generating a sequence of shots. Since an exercise drill could be generated by a variety of shots, an optimal solution for the user-defined target cost was searched in the solution space. Note that the target goal of the optimizer is to fit an exercise drill to user-defined exercise objectives and intensity levels. A Markov chain Monte Carlo optimization method, known as simulated annealing [39], with a Metropolis-Hastings state searching step [12] was used to optimize the exercise drill. To employ the optimization method, a Boltzmann-like objective function was defined:

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

where t denotes the temperature parameter of the simulated annealing process [39], set to decrease gradually during the optimization process. At every iteration, the optimizer chooses and applies a move to the current exercise drill E to propose an exercise drill E' . Based on the three components of the shot (position, speed, and frequency), seven different moves were developed to be chosen by the optimizer:

- change position;
- change speed;
- change frequency;
- change position and speed;
- change position and frequency;
- change speed and frequency; and
- change position, speed, and frequency.

At the beginning of the optimization process, the number of shots defined by the user is generated through random parameters of position (p), speed (v), and frequency (φ). At each iteration of the optimization, one of the shots and one of the moves are selected randomly. Then the move is applied to the selected shot and to the current exercise drill E to create a new exercise drill E' . For example, when the “change position” move is selected, a randomly chosen shot moves to a new position, and the system computes the cost of the new exercise drill E' . In our implementation, the selection probabilities of the moves were set to $Pr_p = .25$ for “change position,” $Pr_v = .25$ for “change speed,” $Pr_\varphi = .25$ for “change frequency,” $Pr_{p,v} = .10$ for

“change position and speed,” $Pr_{p,\phi} = .05$ for “change position and frequency,” $Pr_{v,\phi} = .05$ for “change speed and frequency,” and $Pr_{p,v,\phi} = .05$ for “change position, speed, and frequency.” Our optimization favors the individual changes for position, speed, and frequency. To decide whether to accept a proposed exercise drill E' , our method compares the proposed total cost $C_{Total}(E')$ to the current total cost of $C_{Total}(E)$. The developed method accepts the proposed exercise drill E' based on the Metropolis criterion [12] denoted as:

$$Pr(E'|E) = \min\left(1, \frac{f(E')}{f(E)}\right) \quad (9)$$

To optimize different exercise drill design solutions, the simulated annealing method was employed. A temperature parameter t is first defined. When optimization begins, the temperature parameter t is represented by a high value, allowing the optimizer to aggressively explore the optimized results. As the iterations of the optimization evolve, the temperature parameter is reduced until it reaches zero. An initial temperature $t = 1.0$ was used in the current implementation at the beginning of the optimization and was reduced by .10 every 200 iterations. As the temperature parameter decreases, the optimizer becomes more greedy in finding the optimal solutions. The optimization process is completed when the total cost change is less than 5% in the past 50 iterations. Figure 2 illustrates how the total cost $C_{Total}(E)$ changes over several iterations.



Figure 2: Total cost changes as the optimization process (iterations) evolves. © [2020] IEEE

Unless otherwise specified by a user, the weights assigned to the cost terms responsible for the shots are set to $w_S^{Dist} = 1.0$, $w_S^{Speed} = 1.0$, and $w_S^{Freq} = 1.0$, and the weights of the prior cost terms are set to $w_P^{Dur} = .10$, $w_P^{Var} = .50$, and $w_P^{Side} = .10$. Note that the user is allowed to control the weight values of both shot and prior cost terms to synthesize exercise drills by prioritizing differently the objectives of the drill. Moreover, the user is allowed to control the target values of the cost terms so that he/she can synthesize exercise drills with different levels of difficulty, intensity, and variability. Variations in exercise drills generated by the presented system, while keeping both the target values and weights constant, are shown in Figure 3, whereas shots of various distributions based on different target values are shown in Figure 4.



Figure 3: Variations in exercise drills generated by the presented system through keeping the target exercise amounts ($\sigma_D = .6$, $\sigma_V = .3$, and $\sigma_\Phi = .2$) constant. Numbers close to balls denote the sequence of shots. Note that for all four examples shown in this Figure, the weights of the shot and prior cost terms remained constant. © [2020] IEEE

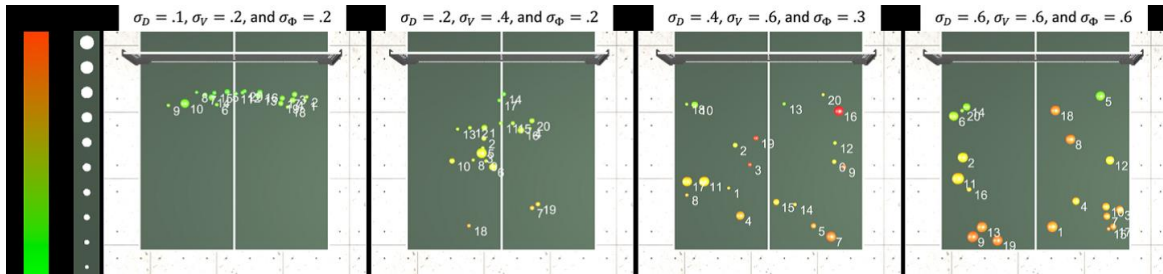


Figure 4: Distribution of shots when varying the target values (σ_D , σ_V , and σ_Φ) of the shots cost terms. Numbers close to balls denote the sequence of shots. Note that for all four examples shown in this figure the weight of the shot and prior cost terms remained constant. © [2020] IEEE

1.4 Evaluation as An Exercise Tool

This study investigates whether the developed virtual reality table tennis application can be used also as an exercise tool. Specifically, this study was conducted to evaluate whether our system can synthesize exercise drills that fulfill user-defined exercise targets. In this study, we considered intensity variations of the exercise drill and we captured heart rates to investigate whether exercise intensity differences are expressed as heart rate differences.

1.4.1 Participants

Participants were recruited through class announcements and emails. The participant group was comprised of 36 healthy undergraduate and graduate students. None of the participants were athletes. The students' ages ranged from 19 to 26 years, with a mean of $M = 22.31$ ($SD = 2.65$). All participants had prior experience with virtual reality; however, none of the participants had experience with exercise sports games in virtual reality. No compensation was given to the students for their participation.

1.4.2 Setup and Implementation Details

The research team conducted this study at a lab space of our university. The lab space was 9 meters long and 7 meters wide, with a ceiling height of 4 meters. All tables and chairs were removed. The HTC Vive Pro head mounted display device was used to project the virtual reality content, and an HTC table tennis racket⁶ and HTC Vive tracker were used to control the virtual racket in the virtual environment. The virtual reality gaming application was developed in the Unity3D game engine version 2019.1.4 and ran on a Dell Alienware Aurora R7 desktop computer (Intel Core i7, NVIDIA GeForce RTX 2080, 32GB RAM). Note that the time required by our system to optimize an exercise drill that consists of 40 shots did not exceed 5 seconds.

A 3D virtual environment of a sports court was designed in 3Ds Max and imported into Unity3D (see Figure 5). The dimensions of the court were identical to those of the lab space. This was done so that the participants would be aware of their position in the real environment in order to eliminate potential accidents that might have occurred by colliding with the walls. The table tennis table was placed in the middle of the room, so the participants were at least three meters

⁶ <https://www.vive.com/us/VR-racket-sports-set/>

away from either wall. Finally, a Polar OH1+ optical heart rate sensor was used to capture the heart rates of the participants to determine how exercise levels of different difficulty, intensity, and variability affected them.



Figure 5: The virtual table tennis court that was designed and used for the purpose of the study.
© [2020] IEEE

1.4.3 Conditions of the Study

Three conditions were tested to determine whether the developed virtual reality gaming application could be used as an effective exercise tool. Note that this is a between-group study, which means that all participants experienced all of the three developed conditions. The conditions were:

- **Low Intensity:** The user does not need to move much, the balls move with low speed, and the shots are generated with a low frequency. The target values were set as: $\sigma_d = .2$, $\sigma_v = .2$, and $\sigma_\phi = .2$. Based on the set target values, the heart rate of participants was expected to reach 110 beats per minute (BPM)).
- **Medium Intensity:** The user is called to perform small steps to hit the ball, the balls move a bit faster, and the shots are generated with a medium frequency. The target values were set as: $\sigma_d = .5$, $\sigma_v = .5$, and $\sigma_\phi = .5$. In this condition, the heart rate of participants was expected to reach 120 bpm.
- **High Intensity:** The user is called to perform more intense movements to hit the ball, the balls move even faster, and the shots are generated with a high frequency.

The target values were set as: $\sigma_d = .8$, $\sigma_V = .8$, and $\sigma_\phi = .8$. The heart rate of participants was expected to reach 130 bpm.

Finally, we would like to mention that for all three examined conditions the weights assigned to the shot and prior cost terms remained constant. By changing the target values of each shot-related cost term, we were able to generate on demand an exercise drill with different objective goals and consequently with different target exercise intensity levels expressed through heart rate indicators. Thus, each of the abovementioned conditions made requests to our optimizer in terms of distance, speed, and frequency objectives, and ultimately specified the intensity level for each synthesized table tennis exercise drill.

1.4.4 Measurements

To evaluate the prospect of using a virtual reality table tennis application as an exercise tool, the heart rate of the participants was measured using the abovementioned heart rate sensor. Specifically, the mean heart rate of participants was computed after each trial of the exercise segment of the study. Note that high heart rate values correspond to higher fatigue [59][62]. For each condition (there were 10 trials in total for each intensity level), the first 30 seconds of the heart rate data were deleted, as this was considered a warm-up period. After the exercise segment of the study the participants were asked to provide a rating of perceived exertion (RPE), as developed by Borg [9]. The RPE scale measures the intensity of an exercise by asking participants to rate the perceived intensity of an activity. We used a seven-point scale in which 1 indicated “not high at all” and 7 indicated “very, very high.”

1.4.5 Procedure

We followed a within-group study design, and we asked the participants to partake in a three-day session. The participants experienced a different condition on each day of the study. Note that the participants were aware of this process before the beginning of the first session. Once the participants arrived at the lab space, the research team provided information about the project, and the participants were asked to sign a consent form that was approved by the Institutional Review Board of our university. During that time, the participants became aware that they would

be attending three sessions. Then, the participants were asked to complete a demographic questionnaire. In the next step, the research team helped the participants with the virtual reality equipment.

Once the virtual reality gaming application started, the research team asked the participants to move to a position close to the table tennis table within the virtual environment. The participants were also told that the walls of the virtual space corresponded to the walls of the real space and that they should be careful when moving toward any of the walls. None of the participants collided with any of the walls during the study. When the participants indicated that they were ready, the researcher switched on the virtual ball-throwing machine. In total, the participants experienced 10 variations of the game at the same intensity condition classification. We developed 10 variations for each condition to ensure that the participants did not lose interest while playing the exergame.

For each variation of the training session, the participants were exposed to a trial in which 40 virtual balls were placed in the ball-throwing machine. Each variation of the condition (trial) lasted no more than two minutes. Note that in between the trials, participants were allowed to take up to a two-minute break. To eliminate the first-order carry-over effects between the trials of the condition, Latin squares [38] for balancing were used. Thus, each participant experienced a different sequence of each of the variations. After the end of the exercise segment, the participants were asked to fill in the perceived exertion scale. At the end of the final trial (day three of the study), the participants were informed that the research team would answer questions about the study. To standardize the study, each participant came to the lab on the same day and time respectively for each of the three sessions that occurred during three consecutive weeks. The Latin squares [38] ordering method was used to ensure a balance across all conditions (low, medium, and high intensity levels). Thus, each participant experienced a different level of exercise intensity in each of the visits. Finally, we would like to note that each participant spent no more than a total of 120 minutes in the lab for the total of the three sessions (40 minutes per session).

1.4.6 Results

A one-way repeated measuring analysis of variance (ANOVA) was used to determine the differences across the three developed intensity levels of the exercise. The normality assumption of the collected data was evaluated graphically using Q-Q plots of the residuals [24]. The Q-Q plots indicated that the obtained data fulfilled the normality criteria. A $p < .05$ value was deemed

statistically significant. Boxplots of the obtained results are shown in Figure 6. By analyzing the heart rate data, we identified significant differences across the three examined conditions [$\Lambda = .189, F(2,34) = 72.900, p < .0001, \eta_p^2 = .811$]. Post-hoc comparisons using the Bonferroni correction revealed that the mean heart rate during the low intensity condition ($M = 108.41, SD = 6.48$) was significantly lower than that of the medium intensity condition ($M = 118.08, SD = 4.87$) at the $p < .0001$ level and that of the high intensity condition ($M = 125.58, SD = 4.81$) at the $p < .0001$ level. Moreover, the mean heart rate during the medium intensity condition was significantly lower than that during the high intensity condition at the $p < .001$ level.

By analyzing the RPE data, we also identified significant differences across the examined conditions [$\Lambda = .616, F(2,34) = 10.576, p < .0001, \eta_p^2 = .384$]. Post-hoc comparisons using the Bonferroni correction revealed that participants reported that the low ($M = 1.89, SD = .82$) and medium ($M = 1.94, SD = .86$) intensity conditions were less intense than the high ($M = 3.05, SD = 1.45$) intensity condition, both at the $p < .0001$ level. No significant difference was found between the low and medium intensity conditions.

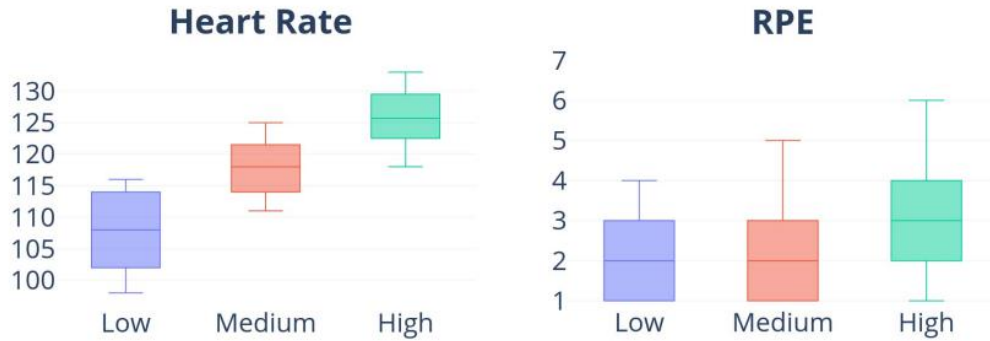


Figure 6: Boxplots of the obtained results regarding the usage of the developed method as an exercise tool. Boxes enclose the middle 50% of the data. The median is denoted by a thick horizontal line. © [2020] IEEE

1.4.7 Discussion

The heart rate of the participants differed across the three exercise levels and, therefore, the three exercise intensity levels worked as expected. Moreover, the heart rate data revealed that it is indeed possible to develop racket sports-related virtual reality gaming applications that can be

used for exercise purposes. The developed exercise design approach also revealed that it can automatically generate table tennis exercise drills that can allow users to exercise at user-specified intensity levels in the comfort of their own living rooms.

Specifically, the collected data has shown that our system can optimize exercise drills that are able to trigger the heart rate of participants close to the target heart rate value that was anticipated for each condition. For instance, the target heart rate value for the low intensity was expected to be 110 BPM and the mean heart rate value of the participants was found to be 108.41 BPM, the target heart rate value for the medium intensity was expected to be 120 BPM and the mean heart rate value of all participants was found to be 118.08 BPM, and the target heart rate value for the high intensity was expected to be 130 BPM and the mean heart rate value of all participants was found to be 125.58 BPM. Based on these findings, it can be said that participants' mean heart rate was closer to the target heart rate when exposed to the low and medium intensity exercise drills compared to the high intensity exercise drill. We believe that the difference between the expected and the actual heart rate could be adjusted by synthesizing exercise drills that take into account the gender, age, and physical health of the users. However, all things considered, our findings indicate that users who are willing to exercise from the comfort of their living rooms while being exposed to a fun activity, such as playing a virtual reality game, can indeed achieve exercise goals significantly close to their desired ones. Finally, we would like to mention that although VRT can help people improve their performance, it does not provide a training experience similar to a real-world one. However, this fact does not invalidate the ability of VRT to help racket sports enthusiasts improve their skills.

Regarding the self-reported intensity of the exercises, on the one hand, we found that the perceived difficulty of the easy and medium intensity drills was lower than that of the high intensity drill. This finding shows that participants were only partially able to distinguish across the intensity levels they were exposed to, since we were not able to find differences between the low and medium intensities. A possible explanation is that the intensity level of an exercise drill can be related to various other participant-related factors (e.g., someone who spends 2-3 days per week at a gym might rate the medium intensity as easy compared to someone who barely exercises). However, each user can tune the intensity of the exercise by triggering the respective target values of the shots cost terms. This way, the developed method can be used to generate exercise drills even for demanding tasks and users who want to exercise at a more advanced level. However, the

fact that we were not able to find differences between the low and medium intensities is the most interesting result of this study. This finding indicates that while the medium workout resulted in a significantly higher heart rate over the low workout, it did not result in a significantly higher perceived exertion. From our point of view, this finding indicates that while the level of exercise intensity increased, the discomfort of participants stayed at low levels.

After the study, we asked participants about their experiences. Almost all of the participants said that they really enjoyed exercising in a virtual reality environment, and many said they liked the way the virtual reality application was designed. We believe that the participants' levels of enjoyment were the reason they did not rate the medium intensity level workout with a higher RPE. This insight (increased levels of enjoyment) could be considered in implementing exercise-related virtual reality applications as a guideline for raising the level of exercise without raising discomfort. Another participant said that including a virtual coach or an opposing player might have also been interesting. However, for the purpose of this study, the motion and the presence of an opposing player might have distracted the user. Finally, it is worth noting that none of the participants reported dizziness or any form of cybersickness.

1.5 Evaluation as A Training Tool

A second study was conducted to evaluate the usefulness of the developed virtual reality gaming application as a training tool. For this evaluation process, the performance of our participants was evaluated in three training conditions: no training, virtual reality training, and real-world training. Details on this study are given in the below subsections.

1.5.1 Participants

The participants were recruited through class announcements and emails. The participant group was comprised of 42 healthy undergraduate and graduate students. It should be noted that none of the participants were athletes. The participants' ages ranged from 19 to 29 years, with a mean of $M = 23.75$ ($SD = 2.64$).

All participants had prior experience with virtual reality. None of the participants of this study had participated in the first study, so they were unaware of the gaming application and its mechanics. No compensation was given for participation. All participants signed a consent form

that was approved by the Institutional Review Board of our university. The participants were randomly divided into three groups: 1) the no training (NT) group, the group of participants that did not receive either virtual reality or real-world training; 2) the virtual reality training (VRT) group, the group of participants that received virtual reality table tennis training using our application; and 3) the real-world training (RWT) group, the group of participants that received real-world table tennis training in our recreation center. It should be noted that each group was comprised of an equal number of participants ($N = 14$, nine males and five females in each group). As this study was divided into multiple sessions (details are given in the next section), the consent form and demographic questionnaire were administered during participants' first visits.

1.5.2 Study Details

This study attempted to determine whether the developed virtual reality gaming application could be used for training purposes. To this end, it measured whether the performance of the participants was improved after participating in virtual reality training sessions. In addition, this study attempted to investigate whether the performance of participants exposed to virtual reality training differs from the performance of participants exposed to real-world training, or no training. The RWT condition was added to investigate whether virtual reality training differs from real-world training. The no training condition was included since we realized that some training would take place during the initial performance evaluation in the recreation center. Thus, this initial assessment alone might have inadvertently led to an improvement in the post-training evaluation. For this reason, we decided to include the no training condition to investigate whether the changed performance of the VRT group depended on the initial performance evaluation or on the actual virtual reality training.

This study is divided into three parts for the VRT and RWT conditions, and into two parts for the NT condition. For all conditions, all participants were asked to attend two 30-minute sessions (a pre and a post-training session) at the table tennis court at the recreation center. The pre- and post-training performance assessment of the participants was performed using a medium-intensity exercise drill. During the performance evaluation sessions, the participants were free to take multiple breaks if needed. For the performance evaluation process, the participants were exposed to 10 trials in which 20 balls were placed in the ball-throwing machine. The machine used

was the iPong V300⁷ table tennis ball-throwing machine. The total duration of the performance evaluation process lasted 30 minutes, including short breaks that the participants took between the study trials. Note that the table tennis table at the recreation center was located in a space of 7 meters long by 4 meters wide.

To evaluate the performance of the participants two measurements were captured: 1) lost shots (the number of balls each participant was unable to hit with the table tennis racket), and 2) mistakes (the number of balls that were either stopped by the net on the table or bounced outside the table after the player hit the ball). Note that since our primary intention was to evaluate the potential improvement in participants' performance, our computations focused on the difference (subtraction) between the post- and pre-training data collected for each participant, which were later used for our statistical analysis process. The participants that were assigned the NT condition did not receive any further training. They were only asked to attend the final performance evaluation session. However, the participants that were assigned the VRT condition were asked to participate in three training sessions, each for the duration of 30 minutes. This group of participants trained using the developed virtual reality table tennis application, and the training sessions were conducted in our department's lab space. During the first two sessions the participants were trained on low-intensity table tennis exercise drills, and during the third session the participants were trained on a medium-intensity exercise drill (see Section 4 for more details on the low- and medium-intensity exercise drills).

The participants that were assigned to the RWT group took part in three training sessions at the recreation center where they trained in real-world table tennis. The participants of the RWT condition were trained on low-intensity table tennis exercise drills during the first two sessions, and on a medium-intensity exercise drill during the third session. The duration of each session was 30 minutes, including short breaks. A table tennis expert (the coach of a local table tennis club; male, 44 years old with 18 years of experience as a table tennis coach) helped us tune the ball-throwing machine (iPong V300) that was used at the recreation center. The tuning of the ball-throwing machine was performed by the expert after experiencing multiple trials of both low and medium exercise intensities in our virtual reality table tennis application.

In conclusion, we would like to note that all participants were aware that they would be attending multiple training sessions before they began the first session; this was made clear when

⁷ <http://www.ipong.net/joomla/index.php/ipong-models/ipong-v300>

the consent form was handed to them. Moreover, all participants followed the same scheduled day pattern: pre-training performance evaluation on Tuesday, first training session on Thursday, second training session on the following Tuesday, third training session on Thursday, and a post-training performance evaluation on the final Tuesday. Each participant came to the lab or recreation center at the same time on each scheduled day. This scheduling process helped us to standardize the study. Finally, regarding the NT condition we would like to note the following. Since participants were asked to take part in only two sessions, we tried to standardize the gap between these sessions. Thus, considering that the last training sessions on both the VRT and RWT were on Thursday and the post-training evaluation session was on Tuesday (a five-day gap), we decided that the performance evaluation session for the NT condition should also be five days after the initial performance evaluation session.

1.5.3 Results

We used the one-way between-group ANOVA to compare the performance of our participants. We used the three conditions (NT, VRT, and RWT) as our independent variables and the performance improvement on lost shots and mistakes (the difference between post- and pre-training scores) as our dependent variables. The obtained results are summarized in Figure 7. Since we conducted a between-group study, before analyzing our results we decided to explore the homogeneity of our participants by using as dependent variables the three participant groups (the three conditions), and as independent variables the age, height, and weight of the participants. Note that, as mentioned before, each group was comprised of nine males and five females. The one-way between-group ANOVA indicated no significant difference for age [$F(2,39) = .901, p = .421$], height [$F(2,39) = 1.188, p = .137$], or weight [$F(2,39) = 1.463, p = .254$] across groups. Based on the obtained results and the male/female ratio per group, it can be said that all three groups were homogenous.

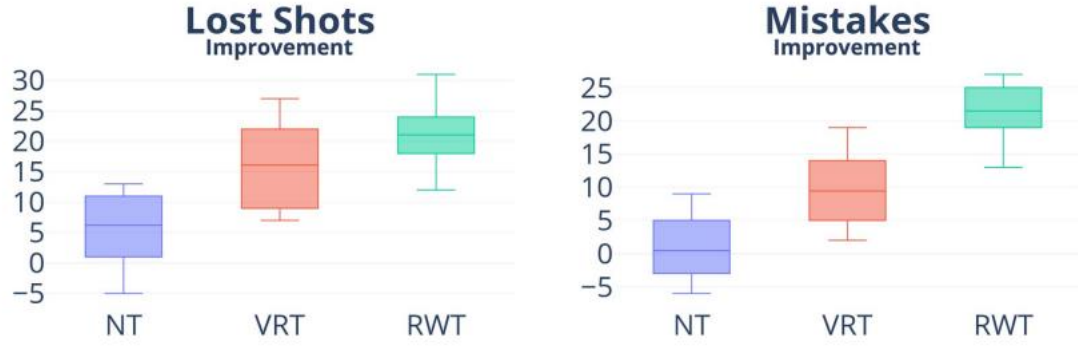


Figure 7: Comparison across the three examined conditions (NT: no training, VRT: virtual reality training, and RWT: real-world training) used to evaluate the developed virtual reality table tennis application. Negative and low values indicate that our participants' performance did not improve at the post-training session. Positive values indicate that participant performance improved. Boxes enclose the middle 50% of the data. The median is denoted by a thick horizontal line. © [2020] IEEE

By analyzing the performance improvement data for lost shots, we found significant differences across the three examined conditions [$F(2,39) = 21.804, p < .001$]. Post-hoc comparisons using the Bonferroni corrected estimates revealed that the mean performance improvement during the NT condition ($M = 6.43, SD = 5.62$) was significantly lower than that of the VRT condition ($M = 14.64, SD = 7.19$) at the $p < .002$ level, and that of the RWT condition ($M = 21.35, SD = 4.92$) at the $p < .001$ level. Moreover, we also found that the performance improvement for the lost shots during the VRT condition were significantly lower than that of the RWT condition at the $p < .05$ level.

By analyzing the performance improvement data for mistakes, we identified significant differences across the three examined conditions [$F(2,39) = 54.824, p < .001$]. Post-hoc comparisons using the Bonferroni corrected estimates showed that the mean performance improvement of mistakes during the NT condition ($M = 1.71, SD = 4.54$) was significantly lower than that of the VRT condition ($M = 9.35, SD = 5.77$) at the $p < .001$ level, and that of the RWT condition ($M = 21.28, SD = 4.53$) at the $p < .0001$ level. Moreover, we also found that the mean performance improvement of mistakes for the VRT condition was significantly lower than that of the RWT condition at the $p < .001$ level.

1.5.4 Discussion

The measurements of lost shots and mistakes revealed that our method can automatically generate training drills that can help table tennis enthusiasts improve their skills. By comparing NT with VRT it can be said that by attending the VR training sessions participants were able to improve their scores on lost shots. This indicates that the training sessions taught that participant group to anticipate the ball and react more appropriately when the ball approached them. Similarly, the participant group that was exposed to virtual reality training had a reduced number of mistakes. This may also be a result of the first finding (reduction of lost shots). Because the participants were anticipating a shot, they were better prepared to react and hit the ball appropriately. Therefore, they were able to perform better overall after receiving the virtual reality table tennis training.

Significant results were also found for both lost shots and mistakes measurements when comparing the VRT and RWT groups. These significant differences are perhaps the most interesting result of the study. The significant results indicate that the RWT group improved their performance even more than the VRT group. It is shown that even if this virtual reality table tennis can be used as a training tool, it is still less effective than RWT. However, considering that we also found a significant improvement in participants' performance when compared to the NT condition, virtual reality training can also be considered as an option for table tennis enthusiasts who wish to improve their skills without having to search for a table tennis coach or partner, or attend training sessions at a gym. Although we found that the VRT training results were lower than the RWT results, it could be said that virtual reality training is still a reasonable and alternative way to train for various reasons. First, people can learn how to play table tennis and improve their skills without needing to actually be in the gym. Second, the virtual reality training for racket sports can be done remotely, saving time and money. Third, virtual reality offers an immersive experience that promotes repetition and retention. It is for these reasons we believe that virtual reality training for sports has multiple potentials for peoples' health and well-being.

Upon completion of the study, we asked participants about their experiences. All the comments we received for the table tennis virtual reality application were quite positive, suggesting that virtual reality table tennis could become a training tool for racket sports enthusiasts. Several participants said that they were not expecting the use of virtual reality to help them improve their skills and reduce the number of mistakes they made. Please note that after the end of the post-training sessions all participants were informed of their performance. Almost all the participants

said that the virtual reality training helped them to become more vigilant. Others said that the virtual reality training helped them better position themselves in the court and that the virtual reality training helped them to react more quickly. Finally, it is worth noting that none of the participants reported dizziness or any form of cybersickness.

1.6 Limitations

There are a few limitations that we would like to note. First, the weight and balance of the table tennis racket that was used in these studies may be problematic. A table tennis paddle weighs between 150 g and 250 g. However, the HTC table tennis racket is 226 g, and with a Vive tracker (89 g) attached it weighs 315 g. Although this weight differential may not be as significant for beginners, we assume that it could be problematic for more experienced and professional table tennis players. We believe that experimentation with fabricating and printing 3D gaming interfaces [70] might solve this problem. Moreover, because the science and technology related to virtual reality is rapidly evolving, we assume that future table tennis paddles for virtual reality experiences will be more similar to actual paddles in terms of weight specifications.

A second limitation is related to missing tactile feedback. Tactile feedback is an important factor that provides an additional parameter to consider when hitting balls [17][51]. To overcome this issue, the use of a tactile actuator might partially solve the tactile feedback problem; however, such a device would add additional weight to the table tennis paddle.

A third limitation is related to the physical space required for a player to use the virtual reality gaming application. Although, as mentioned above, virtual reality racket sports can be experienced from the comfort of one's living room, a large play area with no obstacles would be required in order to achieve an optimal experience without injuring oneself or damaging objects within the real space.

A fourth limitation is that only participants with little or no table tennis experience participated in this study. Unfortunately, we were not able to recruit intermediate or advanced table tennis players. However, it would have been interesting to evaluate our exergame with more experienced table tennis players so we could explore whether the findings of our studies would also apply to them.

The last limitation we would like to mention is related to the head-mounted display that was used for the studies, as some participants commented on it. Specifically, we used the HTC

Vive Pro headset. Participants mentioned that the combination of the wire that connects the headset to the computer, the weight, and the size of the headset itself all made them more concerned and uncomfortable when moving around during game play. We assume that a wireless head-mounted display such as the Oculus Quest might provide a better exergame experience to virtual reality users.

1.7 Conclusions and Future Work Directions

We developed a virtual reality racket sports gaming application and a method for synthesizing exercise drills. In this method, the user may change the parameters of the cost terms and our system will automatically generate an exercise drill that meets these user-specified objectives. Two user studies were conducted to evaluate the effectiveness of the developed application as an exercise and training tool for the table tennis application. The results indicate that virtual reality can be a solution for users who would like to exercise and achieve specified exercise goals. The virtual reality gaming application can also be used to improve user skills. Lastly, the flexibility of the developed gaming application in handling different types of racket sports with minimal changes and effort is another advantage.

Future work might include the development of other virtual reality gaming applications that can be used for exercise and training purposes. An example might be martial arts exercises and training in virtual reality, in which the user could interact with an intelligent virtual coach. In addition to the algorithmic development that automates the exercise and training sessions, exploring the effects of virtual reality exercising and training on long-term health benefits, such as weight loss and rehabilitation, is another interesting direction for future research. Considering the popularity and attractiveness of virtual reality, we can assume that it will be used in the future as a tool that not only entertains users, but also helps them improve their physical functions and overall health.

1.8 References

- [1] Al-Kodmany, Kheir. 2002. Visualization tools and methods in community planning: from freehand sketches to virtual reality. *Journal of planning Literature*, 17(2), 189-211. doi: 10.1177/088541202762475946. Retrieved from [Visualization Tools and Methods in Community Planning: From Freehand Sketches to Virtual Reality \(elibrary.ru\)](http://elibrary.ru/Visualization_Tools_and_Methods_in_Community_Planning:_From_Freehand_Sketches_to_Virtual_Reality)

- [2] Antoun, Rob. 2007. *Women's Tennis Tactics*. Human Kinetics. Retrieved from [Women's Tennis Tactics \(April 23, 2007 edition\) | Open Library](#)
- [3] United States Tennis Association. 1996. *Tennis tactics: winning patterns of play*. Human Kinetics. Retrieved from [Tennis tactics : winning patterns of play : Free Download, Borrow, and Streaming : Internet Archive](#)
- [4] Bailey, W. Bruce and McInnis, Kyle. 2011. Energy cost of exergaming: a comparison of the energy cost of 6 forms of exergaming. *Archives of pediatrics & adolescent medicine*, 165(7), 597-602. doi:10.1001/archpediatrics.2011.15. doi:10.1001/archpediatrics.2011.15. Retrieved from [Energy Cost of Exergaming: A Comparison of the Energy Cost of 6 Forms of Exergaming | Adolescent Medicine | JAMA Pediatrics | JAMA Network](#)
- [5] Banos, Rosa. M., Escobar, Patricia., Cebolla, Ausias., Guixeres, Jaime., Alvarez Pitti, Julio., Lisón, Juan. Francisco and Botella, Cristina. 2016. Using virtual reality to distract overweight children from bodily sensations during exercise. *Cyberpsychology, Behavior, and Social Networking*, 19(2), 115-119. Retrieved from [Using Virtual Reality to Distract Overweight Children from Bodily Sensations During Exercise | Cyberpsychology, Behavior, and Social Networking \(liebertpub.com\)](#)
- [6] Behm-Morawitz, Elizabeth., Lewallen, Jennifer. and Choi, Grace. 2016. A second chance at health: how a 3D virtual world can improve health self-efficacy for weight loss management among adults. *Cyberpsychology, Behavior, and Social Networking*, 19(2), 74-79. Retrieved from [A Second Chance at Health: How a 3D Virtual World Can Improve Health Self-Efficacy for Weight Loss Management Among Adults | Cyberpsychology, Behavior, and Social Networking \(liebertpub.com\)](#)
- [7] Benko, U. and Lindinger, S. 2007. Differential coordination and speed training for tennis footwork. *Coaching and Sport Science Review*, 41, 10-11. Retrieved from [Microsoft PowerPoint - # TENNIS EUROPE 2008 prs Coordination2-video \(cztenis.cz\)](#)
- [8] Biddiss, Elaine. and Irwin, Jennifer. 2010. Active video games to promote physical activity in children and youth: a systematic review. *Archives of pediatrics & adolescent medicine*, 164(7), 664-672. doi:10.1001/archpediatrics.2010.104. Retrieved from [Active Video Games to Promote Physical Activity in Children and Youth: A Systematic Review | Adolescent Medicine | JAMA Pediatrics | JAMA Network](#)
- [9] Borg, Gunnar. A. 1982. Psychophysical bases of perceived exertion. *Medicine & science in sports & exercise*. Retrieved from [Psychophysical bases of perceived exertion. \(apa.org\)](#)
- [10] Brunnett, G., Rusdorf, S. and Lorenz, M. 2006. V-Pong: an immersive table tennis simulation. *IEEE Computer Graphics and Applications*, 26(4), 10-13. doi: 10.1109/MCG.2006.92
- [11] Chhikara, Raj S. 1988. *The inverse Gaussian distribution: theory: methodology, and applications* (Vol. 95). CRC Press.

- [12] Chib, Siddhartha. and Greenberg, Edward. 1995. Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4), 327-335.
- [13] Choi, Woohyeok., Oh, Jeungmin., Edge, Darren., Kim, Joohyun. and Lee, Uichin. 2016. SwimTrain: exploring exergame design for group fitness swimming. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (pp. 1692-1704).
- [14] Compton, Kate., & Mateas, Michael. 2006. Procedural level design for platform games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (Vol. 2, No. 1, pp. 109-111).
- [15] Cotman, Carl. W. and Berchtold, Nicole. C. 2002. Exercise: a behavioral intervention to enhance brain health and plasticity. *Trends in neurosciences*, 25(6), 295-301.
- [16] Cotman, Carl. W., & Engesser-Cesar, Christie. 2002. Exercise enhances and protects brain function. *Exercise and sport sciences reviews*, 30(2), 75-79.
- [17] Davids, K., Bennett, S. I. J. and Beak, S. 2004. Sensitivity of children and adults to haptic information in wielding tennis rackets. In *Interceptive actions in sport* (pp. 211-227). Routledge.
- [18] Deutsch, Judith. E., Brettler, Arielle., Smith, Caroline., Welsh, Jamie., John, Roshan., Guarrera-Bowlby, Phyllis. and Kafri, Michal. 2011. Nintendo Wii sports and Wii fit game analysis, validation, and application to stroke rehabilitation. *Topics in stroke rehabilitation*, 18(6), 701-719.
- [19] Diamond, Adele. 2015. Effects of physical exercise on executive functions: going beyond simply moving to moving with thought. *Annals of sports medicine and research*, 2(1), 1011.
- [20] Donalek, Ciro., Djorgovski, S. G., Cioc, Alex., Wang, Anwell., Zhang, Jerry., Lawler, Elizabeth., Yeh, Stacy., Mahabal, Ashish., Graham, Matthew., Drake, Andrew., Davidoff, Scott., Norris, Jeffrey S. and Longo, Giuseppe. 2014. Immersive and collaborative data visualization using virtual reality platforms. In *2014 IEEE International Conference on Big Data (Big Data)* (pp. 609-614). IEEE.
- [21] Dunlop, John. 2000. Characterising the service bounce using a speed gun. *Tennis science & technology*, 183-190.
- [22] Finkelstein, Samantha., Nickel, Andrea. Suma, Lipps, Zachary., Barnes, Tiffany., Wartell, Zachary. and Suma, Evan. A. 2011. Astrojumper: Motivating exercise with an immersive virtual reality exergame. *Presence: Teleoperators and Virtual Environments*, 20(1), 78-92.
- [23] Gerling, Kathrin. and Mandryk, Regan. 2014. Custom-designed motion-based games for older adults: A review of literature in human-computer interaction. *Gerontechnology*, 12(2), 68-80.

- [24] Ghasemi, Asghar. and Zahediasl, Saleh. 2012. Normality tests for statistical analysis: a guide for non-statisticians. *International journal of endocrinology and metabolism*, 10(2), 486.
- [25] Gomez-Pinilla, Fernando. and Hillman, Charles. 2013. The influence of exercise on cognitive abilities. *Comprehensive Physiology*, 3(1), 403.
- [26] Gurusamy, Kurinchi. Selvan., Aggarwal, Rajesh., Palanivelu, Latha. and Davidson, Brian. R. 2009. Virtual reality training for surgical trainees in laparoscopic surgery. *Cochrane database of systematic reviews*, (1).
- [27] Hayes, Elisabeth. and Silberman, Lauren. 2007. Incorporating video games into physical education. *Journal of Physical Education, Recreation & Dance*, 78(3), 18-24.
- [28] Heng, P. A., Cheng, Chun-Yiu., Wong, Tien-Tsin., Xu, Yangsheng., Chui, Yim-Pan., Chan, Kai-Ming. and Tso, Shiu-Kit. 2004. A virtual-reality training system for knee arthroscopic surgery. *IEEE Transactions on Information Technology in Biomedicine*, 8(2), 217-227.
- [29] Horn, Britton., Dahlskog, Steve., Shaker, Noor., Smith, Gillian. and Togelius, Julian. 2014. A comparative evaluation of procedural level generators in the mario ai framework. In *Foundations of Digital Games 2014*, Ft. Lauderdale, Florida, USA (pp. 1-8). Society for the Advancement of the Science of Digital Games.
- [30] Hoskins-Burney, Tina. and Carrington, Lex. 2014. *The tennis drill book*. Human Kinetics.
- [31] IJsselsteijn, Wijnand., De Kort, Yvonne., Westerink, Joyce., De Jager, Markov. and Bonants, Ronald. 2004. Fun and sports: Enhancing the home fitness experience. In *Entertainment Computing-ICEC 2004: Third International Conference*, Eindhoven, The Netherlands, September 1-3, 2004. Proceedings 3 (pp. 46-56). Springer Berlin Heidelberg.
- [32] Isbister, Katherine. 2013. *Motion-Based Game Design: How to Design Quality Movement Games for Wii, Kinect, and Move*. Morgan Kaufmann Publishers Inc..
- [33] Jack, D., Boian, R., Merians, A. S., Tremaine, M., Burdea, G. C., Adamovich, S. V., Recce, M. and Poizner, H. 2001. Virtual reality-enhanced stroke rehabilitation. *IEEE transactions on neural systems and rehabilitation engineering*, 9(3), 308-318.
- [34] Jennings-Teats, Martin., Smith, Gillian. and Wardrip-Fruin, Noah. 2010. Polymorph: A model for dynamic level generation. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (Vol. 6, No. 1, pp. 138-143).
- [35] Jenny, Seth. E., Manning, R. Douglas., Keiper, Margaret. C. and Olrich, Tracy. W. 2017. Virtual (ly) athletes: where eSports fit within the definition of “Sport”. *Quest*, 69(1), 1-18.
- [36] Jensen, Mads. M., Rasmussen, Majken. K., Mueller, Florian. F. and Grønæk, Kaj. 2015. Keepin'it real: challenges when designing sports-training games. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (pp. 2003-2012).

- [37] Karabiyik, Umit., Mousas, Christos., Sirota, Daniel., Iwai, Takahide. and Akdere, Mesut. 2019. A virtual reality framework for training incident first responders and digital forensic investigators. In *Advances in Visual Computing: 14th International Symposium on Visual Computing, ISVC 2019*, Lake Tahoe, NV, USA, October 7–9, 2019, Proceedings, Part II 14 (pp. 469-480). Springer International Publishing.
- [38] Keedwell, A. Donald. and Dénes, Jozsef. 2015. *Latin squares and their applications*. Elsevier.
- [39] Kirkpatrick, S., Gelatt Jr, C. D. and Vecchi, M. P. 1983. Optimization by simulated annealing. *science*, 220(4598), 671-680.
- [40] Kovacs, Mark. S. 2007. Tennis physiology: training the competitive athlete. *Sports medicine*, 37, 189-198.
- [41] Kozlov, Michail. D. and Johansen, Mark. K. 2010. Real behavior in virtual environments: Psychology experiments in a simple virtual-reality paradigm using video games. *Cyberpsychology, behavior, and social networking*, 13(6), 711-714.
- [42] Krohne, Heinz. W. and Hindel, Christoph. 1988. Trait anxiety, state anxiety, and coping behavior as predictors of athletic performance. *Anxiety Research*, 1(3), 225-234.
- [43] Lanningham-Foster, Lorraine., Jensen, Teresa. B., Foster, Randal. C., Redmond, Aoife. B., Walker, Brian. A., Heinz, Dieter. and Levine, James. A. 2006. Energy expenditure of sedentary screen time compared with active screen time for children. *Pediatrics*, 118(6), e1831-e1835.
- [44] Laufer, Yocheved., Dar, Gali. and Kodesh, Einat. 2014. Does a Wii-based exercise program enhance balance control of independently functioning older adults? A systematic review. *Clinical interventions in aging*, 1803-1813.
- [45] Li, Wanwan., Xie, Biao., Zhang, Yongqi., Meiss, Walter., Huang, Haikun. and Yu, Lap-Fai. 2020. Exertion-aware path generation. *ACM Trans. Graph.*, 39(4), 115.
- [46] Lin, Jih-Hsuan. 2015. “Just Dance”: the effects of exergame feedback and controller use on physical activity and psychological outcomes. *Games for health journal*, 4(3), 183-189.
- [47] Littleford, J. and Magrath, A. 2010. Tennis strokes and tactics to improve your game. *A&C Black*.
- [48] MacKenzie, I. Scott. 1993. “Fitts' law as a performance model in human-computer interaction.” PhD dissertation, University of Toronto, Canada.
- [49] MacKenzie, I. Scott. 1995. Movement time prediction in human-computer interfaces. In *Readings in human-computer interaction* (pp. 483-493). Morgan Kaufmann.

- [50] Mailliot, Pauline., Perrot, Alexandra. and Hartley, Alan. 2012. Effects of interactive physical-activity video-game training on physical and cognitive function in older adults. *Psychology and aging*, 27(3), 589.
- [51] Marchal-Crespo, Laura., van Raai, Mark., Rauter, Georg., Wolf, Peter. and Riener, Robert. 2013. The effect of haptic guidance and visual feedback on learning a complex tennis task. *Experimental brain research*, 231, 277-291.
- [52] McMahan, Ryan. P. and Herrera, Nicolas. S. 2016. AFFECT: altered-fidelity framework for enhancing cognition and training. *Frontiers in ICT*, 3, 29.
- [53] McMorris, Terry. E., Tomporowski, Phillip. E. and Audiffren, Michel. E. 2009. *Exercise and cognitive function*. Wiley-Blackwell.
- [54] Mellecker, Robin., Lyons, Elizabeth. J. and Baranowski, Tom. 2013. Disentangling fun and enjoyment in exergames using an expanded design, play, experience framework: a narrative review. *GAMES FOR HEALTH: Research, Development, and Clinical Applications*, 2(3), 142-149.
- [55] Michalski, Stefan. Carlo., Szpak, Ancret., Saredakis, Dimitrios., Ross, Tyler. James., Billinghamurst, Mark. and Loetscher, Tobias. 2019. Getting your game on: Using virtual reality to improve real table tennis skills. *PloS one*, 14(9), e0222351.
- [56] Miller, Joshua. M., Hilbert, Scott. C. and Brown, Lee. E. 2001. Speed, quickness, and agility training for senior tennis players. *Strength & Conditioning Journal*, 23(5), 62.
- [57] Molet, Tom., Aubel, Amaury., Çapin, Tolga., Carion, Stephane., Lee, Elwin., Magnenat-Thalmann, Nadia., Noser, Hansrudi., Pandzic, Igor., Sannier, Gael. and Thalmann, Daniel. 1999. Anyone for tennis?. *Presence: Teleoperators & Virtual Environments*, 8(2), 140-156.
- [58] Monedero, Javier., Lyons, Elizabeth. J. and O’Gorman, Donal. J. 2015. Interactive video game cycling leads to higher energy expenditure and is more enjoyable than conventional exercise in adults. *PloS one*, 10(3), e0118470.
- [59] Mostert, S. and Kesselring, J. 2002. Effects of a short-term exercise training program on aerobic fitness, fatigue, health perception and activity level of subjects with multiple sclerosis. *Multiple Sclerosis Journal*, 8(2), 161-168.
- [60] Mousas, Christos., Anastasiou, Dimitris. and Spantidi, Ourania. 2018. The effects of appearance and motion of virtual characters on emotional reactivity. *Computers in Human Behavior*, 86, 99-108.
- [61] Murray, Craig. D., Patchick, Emma., Pettifer, Stephen., Caillette, Fabrice. and Howard, Toby. 2006. Immersive virtual reality as a rehabilitative technology for phantom limb experience: a protocol. *CyberPsychology & Behavior*, 9(2), 167-170.

- [62] Nelesen, Richard., Dar, Yasmin., Thomas, KaMala. and Dimsdale, Joel. E. 2008. The relationship between fatigue and cardiac functioning. *Archives of internal medicine*, 168(9), 943-949.
- [63] Nelson, Michael., Koiliias, Alexadros., Gubbi, Sahana. and Mousas, Christos. 2019. Within a virtual crowd: Exploring human movement behavior during immersive virtual crowd interaction. In *The 17th International Conference on Virtual-Reality Continuum and its Applications in Industry* (pp. 1-10).
- [64] Ogawa, Elisa. F., You, Tongjian. and Leveille, Suzanne. G. 2016. Potential benefits of exergaming for cognition and dual-task function in older adults: a systematic review. *Journal of aging and physical activity*, 24(2), 332-336.
- [65] Padala, Kalpana. P., Padala, Prasad. R., Lensing, Shelly. Y., Dennis, Richard. A., Bopp, Melinda. M., Roberson, Paula. K. and Sullivan, Dennis. H. 2017. Home-based exercise program improves balance and fear of falling in community-dwelling older adults with mild Alzheimer's disease: a pilot study. *Journal of Alzheimer's disease*, 59(2), 565-574.
- [66] Pasch, Marco., Bianchi-Berthouze, Nadia., van Dijk, Betsy. and Nijholt, Anton. 2009. Movement-based sports video games: Investigating motivation and gaming experience. *Entertainment computing*, 1(2), 49-61.
- [67] Patibanda, Rakesh., Mueller, Florian. F., Leskovsek, Matevz. and Duckworth, Jonathan. 2017. Life tree: understanding the design of breathing exercise games. In *Proceedings of the annual symposium on computer-human interaction in play* (pp. 19-31).
- [68] Pluchino, Alessandra., Lee, Sae. Yong., Asfour, Shihab., Roos, Bernard. A. and Signorile, Joseph. F. 2012. Pilot study comparing changes in postural control after training using a video game balance board program and 2 standard activity-based balance intervention programs. *Archives of physical medicine and rehabilitation*, 93(7), 1138-1146.
- [69] Prohaska, Thomas. R. and Peters, Karen. E. 2007. Physical activity and cognitive functioning: Translating research to practice with a public health approach. *Alzheimer's & Dementia*, 3(2), S58-S64.
- [70] Rajguru, Chinmay., Mousas, Christos. and Yu, Lap-Fai. 2019. Computational design and fabrication of customized gamepads. In *The 17th International Conference on Virtual-Reality Continuum and its Applications in Industry* (pp. 1-11).
- [71] Rao, C. R., Sinharay, S., & Sinharay, S. 2007. *Psychometrics*, Vol. 26. Elsevier.
- [72] Rodrigues, Sergio. Tosi. 2000. "Visuo-motor coordination in table tennis". PhD thesis, University of Calgary.

- [73] Rosenberg, Dori., Depp, Colin. A., Vahia, Ipsit. V., Reichstadt, Jennifer., Palmer, Barton. W., Kerr, Jacqueline., Norman, Greg. and Jeste, Dilip. V. 2010. Exergames for subsyndromal depression in older adults: a pilot study of a novel intervention. *The American Journal of Geriatric Psychiatry*, 18(3), 221-226.
- [74] Ross, Alyson. and Thomas, Sue. 2010. The health benefits of yoga and exercise: a review of comparison studies. *The journal of alternative and complementary medicine*, 16(1), 3-12.
- [75] Saito, Kei., Masai, Katsutoshi., Sugiura, Yuta., Kimura, Toshitaka. and Sugimoto, Maki. 2018. Development of a virtual environment for motion analysis of tennis service returns. In *Proceedings of the 1st International Workshop on Multimedia Content Analysis in Sports* (pp. 59-66).
- [76] Schoene, Daniel., Valenzuela, Trinidad., Lord, Stephen. R. and de Bruin, Eling. D. 2014. The effect of interactive cognitive-motor training in reducing fall risk in older people: a systematic review. *BMC geriatrics*, 14(1), 1-22.
- [77] Schoene, Daniel., Valenzuela, Trinidad., Toson, Barbara., Delbaere, Kim., Severino, Connie., Garcia, Jaime., Davies, Thomas. A., Russell, Frances., Smith, Stuart T. and Lord, Stephen. R. 2015. Interactive cognitive-motor step training improves cognitive risk factors of falling in older adults—a randomized controlled trial. *PLoS one*, 10(12), e0145161.
- [78] Shaw, Lindsay. Alexander. 2014. “Development and evaluation of an exercycle game using immersive technologies”. Bachelor Thesis. The University of Auckland, New Zealand.
- [79] Shi, Peizhi. and Chen, Ke. 2016. Online level generation in Super Mario Bros via learning constructive primitives. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)* (pp. 1-8). IEEE.
- [80] Sinclair, Jeff., Hingston, Philip. and Masek, Martin. 2007. Considerations for the design of exergames. In *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia* (pp. 289-295).
- [81] Sorenson, Nathan. and Pasquier, Philippe. 2010. The Evolution of Fun: Automatic Level Design Through Challenge Modeling. In *ICCC* (pp. 258-267).
- [82] Staiano, Amanda. E., Abraham, Anisha. A. and Calvert, Sandra. L. 2013. Adolescent exergame play for weight loss and psychosocial improvement: a controlled physical activity intervention. *Obesity*, 21(3), 598-601.
- [83] Sveistrup, Heidi. 2004. Motor rehabilitation using virtual reality. *Journal of neuroengineering and rehabilitation*, 1(1), 1-8.
- [84] Van Diest, Mike., Lamothe, Claudine. J., Stegenga, Jan., Verkerke, Gijsbertus. J. and Postema, Klaas. 2013. Exergaming for balance training of elderly: state of the art and future developments. *Journal of neuroengineering and rehabilitation*, 10(1), 1-12.

- [85] Wüest, Seline., Borghese, Nunzio. A., Pirovano, Michele., Mainetti, Renato., van de Langenberg, Rolf. and de Bruin, Eling. D. (2014). Usability and effects of an exergame-based balance training program. *GAMES FOR HEALTH: Research, Development, and Clinical Applications*, 3(2), 106-114.
- [86] Xie, Biao., Zhang, Yongqi., Huang, Haikun., Ogawa, Elisa., You, Tongjian and Yu, Lap-Fai. 2018. Exercise intensity-driven level design. *IEEE transactions on visualization and computer graphics*, 24(4), 1661-1670.
- [87] Zhang, Yongqi., Xie, Biao., Huang, Haikun., Ogawa, Elisa., You, Tongjian and Yu, Lap-Fai. 2019. Pose-guided level design. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (pp. 1-12).

CHAPTER 2. PUBLISHED ARTICLE #2: VIRTUAL REALITY GAME LEVEL LAYOUT DESIGN FOR REAL ENVIRONMENT CONSTRAINTS

All the authors in the paper agreed on the author of the dissertation to use this publication in her Ph.D. dissertation.

This paper was published in Graphics and Visual Computing Journal, presented at CAD & Graphics 2021.

© [2021] Elsevier. Reprinted, with permission, from [Liu, Huimin., Wang, Zhiquan., Mazumdar, Angshuman. and Mousas, Christos, Virtual reality game level layout design for real environment constraints. *Graphics and Visual Computing*, 2021]

(Cite) Liu, H., Wang, Z., Mazumdar, A., & Mousas, C. (2021). Virtual reality game level layout design for real environment constraints. *Graphics and Visual Computing*, 4, 200020.
<https://doi.org/10.1016/j.gvc.2021.200020>

The authors confirm contribution to the paper as follows:

- Study Conceptualization and Design: Mousas, Christos;
- Methodology and/or Simulation Design: Mousas, Christos;
- Execution/Implementation: Liu, Huimin, Wang, Zhiquan and Mazumdar, Angshuman;
- Data collection: Mousas, Christos and Liu, Huimin;
- Analysis and interpretation of results: Mousas, Christos and Liu, Huimin;
- Findings, Conclusions, and/or Recommendations: Mousas, Christos and Huimin Liu;
- Draft manuscript preparation: Mousas, Christos and Liu, Huimin.

All authors reviewed the results and approved the final version of the manuscript.
All authors in this paper agree on the usage of the paper in this dissertation.
Purdue does not require any formal permission.

Abstract

This paper presents an optimization-based approach for designing virtual reality game level layouts, based on the layout of a real environment. Our method starts by asking the user to define the shape of the real environment and the obstacles (e.g., furniture) located in it. Then, by representing a game level as an assembly of chunks and defining the game level layout design decisions in cost terms (mapping, fitting, variations, and accessibility) in a total cost function, our system automatically synthesizes a game level layout that fulfills the real environment layout and its constraints as well as the user-defined design decisions. To evaluate the proposed method, a user study was conducted. The results indicated that the proposed method: (1) enhanced the levels of presence; (2) enhanced the levels of involvement of participants in the virtual environment; and (3) reduced the fear of collision with the real environment and its constraints. Limitations and future research directions are also discussed.

2.1 Introduction

Virtual reality games are designed so that the player uses controllers to navigate in the virtual environment. However, navigation through locomotion is considered one of the universal tasks performed in real and virtual environments [1]. Moreover, sensorimotor actions are essential factors in providing a compelling experience for virtual reality users [2], just as the mismatching between the real and virtual environment is equally essential in impacting the movement behavior and arousal of virtual reality users [3]. If the user cannot naturally move around and engage with the virtual environment as they would in a real one, then the illusion of being in another place would diminish, making the whole play experience poor and less realistic.

Given the fact that walking is a simple and intuitive method of interaction in the environment, providing a game player with the ability to experience the gaming environment by walking in it would likely enhance the player's gaming experience (i.g., the player will be able to freely move and interact in the virtual game level environment). However, it is impossible for a game level designer to know in advance the layout and size of the real environment and the obstacles (e.g., furniture) involved. Therefore, it is difficult to create customized game levels for numerous real environment configurations. To overcome this challenge, this paper presents a novel computational approach that considers both the shape of the real environment and its obstacles,

and automatically generates game levels that take into account the real environment layout and its constraints (see Figure 8).

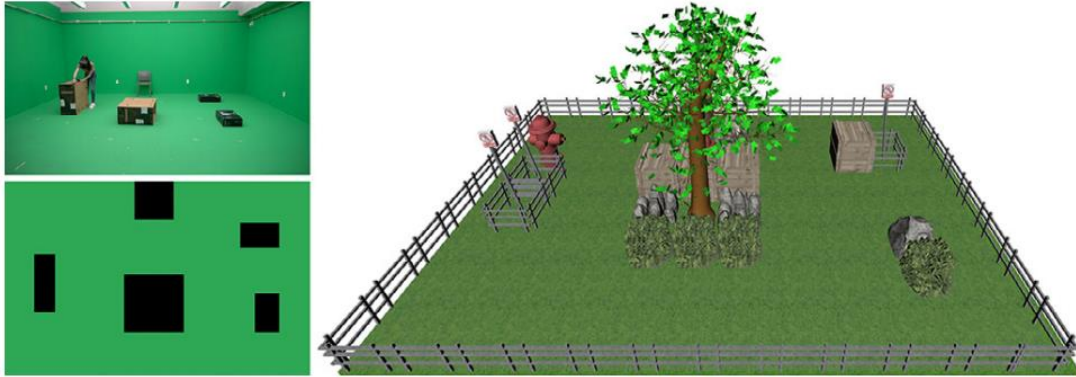


Figure 8: Given a real environment and its constraints, the user can easily capture its layout using a virtual reality controller. Later, our proposed method synthesizes a game level layout that matches the real environment layout and its constraints. © [2021] Elsevier

Our method first considers that a game level can be represented as an assembly of multiple game level chunks. Second, it asks the user to define the play area and the constraints/obstacles located in it by simply using a virtual reality controller (e.g., the Oculus Touch controller); thus, the real environment's layout is generated. Third, our method assigns four cost terms to a total cost function that encodes design decisions (mapping, fitting, variations, and accessibility) and provides a designer the ability to prioritize the cost terms in order to generate level layouts with different objectives. Finally, the game levels are synthesized using an optimization-based method, the Markov Chain Monte Carlo. We implemented our optimization framework as a plug-in for the Unity game engine and demonstrated how it could generate different types of games. We intend to release the plug-in for public use.

To understand the effectiveness of our method, a user study was conducted. The results indicated that the proposed method (1) enhanced the levels of presence, (2) enhanced the levels of involvement of participants in the virtual environment, and (3) reduced the fear of collision with the real environment. The significant contributions of our work include:

- An optimization-based approach to design game level layouts reflecting the real environment and its constraints. Such a concept could be used in a variety of virtual

reality games and therefore improving user experience while also allowing users to immerse themselves in the virtual world and the gaming scenario.

- The ability of our approach to generate game level designs for different layouts and constraints.
- The ability to customize the synthesized game level by prioritizing each cost term.

We think that both the game development and the virtual reality community will benefit from such a method. Our method provides any user the ability to walk and interact in the gaming environment more efficiently, even though in a constrained real environment (e.g., living room). Each space is unique in shape, size, and furniture configuration (there might be empty spaces or spaces with a lot of furniture). Therefore, a method that automatically synthesizes a game level for any real environment is essential to provide all game players with a compelling virtual reality gaming experience.

This paper is organized in the following sections. Section 2.2 covers related work. Section 2.3 presents preliminary information on the proposed method. Section 2.4 describes the way game level design problems are formulated and solved. Section 2.5 outlines the details for implementation. Section 2.6 presents the conducted user study and its results. Finally, Section 2.7 summarizes our conclusions, our method’s limitations, and potential future work.

2.2 Related Work

The difficulty of moving through a highly constrained real environment while observing the virtual environment lies in the impact such an environment has on the sense of presence and immersion of virtual reality users [4][5]; the result could be an experience that is entirely less engaging to the user. Therefore, natural walking in virtual environments remains a challenge primarily because of the large space required to allow the user to experience the virtual world [6]. However, among others, redirected walking [7][8][9] and virtual-to-real environment mapping [10] partially solve the limited space problem of real environments by manipulating the user’s real-world trajectories.

Several attempts have been made to overcome the limitation of experiencing a virtual environment while walking in a real one. Nescher et al. [11] proposed a method that analyzes the real environment in advance in order to identify walkable areas in the virtual environment. Later,

this information is processed and used in order to provide ad hoc free walking in virtual environments when the user finds himself in a constrained virtual environment. Shapira et al. [12] developed a method that analyzes the user's real environment in order to identify flat surfaces, like a wall or a couch, which will determine candidate locations for placing virtual objects in the real environment. Nuernberger et al. [13] developed an augmented reality application in which the edges and planes of the real environment are detected so they can be aligned with virtual content placement. McGill et al. [14] proposed the Augmented Virtuality [14] which adds out-of-context information to the virtual environment only when considered necessary by the system. Other methods include the use of occupancy maps and glass walls [15][16], virtual environment layers [17][18], such as wireframes or other visual indicators that indicate the presence of real obstacles, and vibrotactile actuators in the head-mounted display that trigger an alert signal when users are approaching obstacles [18]. However, in most of the previously mentioned methods the sense of presence was not improved, indicating a significant limitation when experiencing a virtual environment.

Prior research has also focused on using 3D scanning technology to acquire a replica of the real environment. For example, Kanamori et al. [19] used a 3D scanner to scan a real environment and superimposed the 3D point cloud of the user's real environment onto the virtual environment through the head-mounted display. Sra et al. [20] used 3D scanning technology to acquire a 3D model of the real environment. Then, by detecting the walkable area, they could generate fences or water that would prevent the user from walking in specific locations. In addition, by using a small dataset of objects, Sra et al. [20] could substitute real environment objects with virtual ones. Although promising, this approach requires using a 3D scanner, equipment that only a limited number of virtual reality users can access in their homes. Moreover, the applicability of such a method in creating virtual environments for game purposes is unclear.

In this paper, it is proposed the use of an optimization-based method, which has been extensively used in designing virtual environments quickly and affordably [21][22][23][24][25]. The optimization-based synthesis of game-related content is a critical technique used extensively in the modern game-development process [26]. Furthermore, optimization-based design techniques are beginning to enhance game replayability because they offer users the ability to play multiple variations of a single game. Examples of optimization-based methods include the work of Hartsook et al. [27] and Hullett and Mateas [28]. They employed optimization-based design

methods for matters of adaptability or replayability. Such methods also provide the ability to design games that adapt to a variety of constraints and parameters; both during the initialization process and before the game starts [29][30][31]. Optimization-based methods can even alter a game dynamically in response to events in the game [32][33][34][35].

Our method synthesizes a game level layout taking into account the real environments and its constraints, while also allowing a game level designer to control the synthesized gaming environment by prioritizing the cost terms of the presented total cost function. Thus, our method facilitates designer control over generated content and gives players the ability to dictate the degree to which the synthesized game will focus. Our method requires a simple virtual reality controller that comes with a head-mounted display to capture the layout of the real environment. This feature is in contrast to the 3D scanner common in previous methods [19][20]. Although no precise information about the real environment could be captured using a virtual reality controller, our low-cost layout-capturing method provides enough data to sufficiently synthesize the game levels. We think our method could be useful for the automatic design of unique virtual reality game level layouts without the need for additional hardware.

2.3 Preliminaries

This section presents the preliminary steps required to develop our pipeline. The steps include: (1) the capture process for the real environment and its associated constraints; (2) the game level chunks that need to be designed for synthesizing the virtual reality game levels; and (3) generating and characterizing the virtual grid.

2.3.1 Capturing the real environment and Constraints

Our method begins by asking the user to capture the real environment and its constraints in which the virtual reality game will be created. This process generates a virtual layout according to the real environment in which the user is willing to play the game. The player is then asked to use the controller, which comes with the head-mounted display, to define the real environment's entire play area. To do so, the player simply clicks a button when the controller is located at the diagonal corners of the environment. In our case, we are using the Oculus Quest head-mounted display and the Oculus Touch controller (Oculus Quest does not allow passthrough access;

therefore, a user should take slightly off the head-mounted display to observe the real environment during the capturing process). In larger and more complex shaped environments, in which the shape is more than a single rectangle, the user can capture multiple rectangular shapes. Later, these shapes are combined by our system to provide the actual shape of the environment. Note that only the area of the plane (x, z coordinates) is captured.

Next, the user is asked to define the constraints found in the real environment. Constraints are defined as any object that might prevent the user from moving within the physical environment (e.g., coffee tables, couches, television stands, chairs, etc.). For this process we use the virtual reality controller, with which the user must define a shape that encloses the objects. After the user finishes the capturing process for the real environment and its constraints, the system generates the environment's layout. Figure 8 shows examples of the real environment and the associated layouts generated based on the process described above.

This paper does not present a commercial product but a proof of concept; that is, how to automatically design virtual reality game levels based on real environment constraints by considering a number of game level layout design decisions. To simplify the process of capturing the environment and its constraints, we use rectangular shapes. If a number of resources for game level chunks are available, a developer could easily extend our approach to capture the real environment more precisely by incorporating additional shapes or using a paint-based method to define environment's boundaries. However, most virtual reality systems use base stations which enclose the user in a square shaped area. Thus, our proposed method considered only rectangular-shaped game level chunks.

2.3.2 Game level chunks

To synthesize game levels based on a real environment and its constraints, we used game level chunks (see Figure 9). Because this project is a proof of concept, we decided to use a relatively small number of chunks compared to multiple 3D game objects found in a commercial game. However, given the availability of various resources, a developer could easily extend our approach and incorporate more game level chunks. Our particular game level chunks are divided into three types:

- **Open Space Chunks:** Open space chunks are placed in free-from-obstacles grid cells and are used to define the virtual environment's walkable area.

- **Boundary Chunks:** Boundary chunks are placed in the virtual environment's boundaries in order to inform users of the boundaries and prevent them from colliding with the walls in the real environment. We developed two types of boundary chunks: (1) the corner boundary chunk and (2) the straight boundary chunk.
- **Obstacle Chunks:** Obstacle chunks are intended to substitute real environment obstacles in the virtual reality game. These chunks inform the user that particular areas in the virtual environment are occupied, thereby preventing the user from colliding with real environment obstacles.

Each chunk is represented with the label that characterizes it and a directional vector (up, down, left, right, right and up, right and down, left and up, and left and down). The directional vector is later used to correctly align the game level chunks with the generated grid map. Additional chunks might be needed for more complex games and real environments. However, based on our experiments during development, we found that the three types of game level chunks are sufficient enough to cover almost any area and allow developers to synthesize a virtual reality game level that can fit a real environment and its constraints.

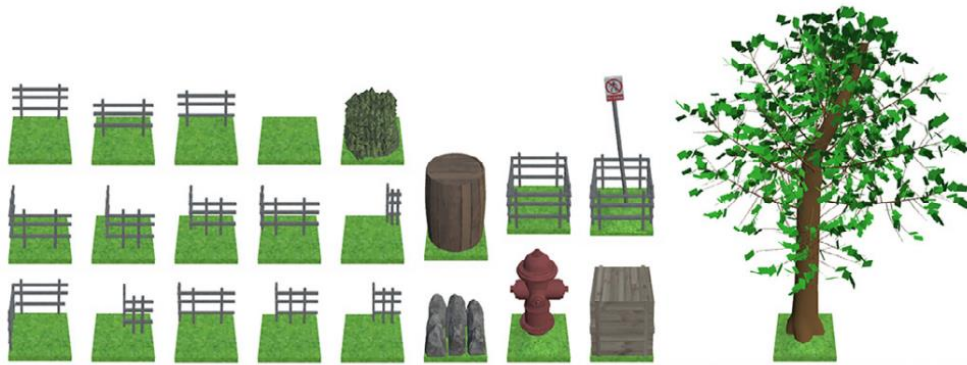


Figure 9: Examples of game level chunks developed for the proposed project. © [2021] Elsevier

2.3.3 Grid generation and characterization

Given the area captured by the user as input, our system generates a $M \times N$ virtual grid, which encloses the entire captured area. Later, a part of this grid is used to synthesize the game

level. For this project, a cell in the grid has a dimension of 50×50 cm, which is equal to the size of each game level chunk; however, other dimensions could also be considered. Our system uses an inside-outside test [36] to identify which grid cells correspond to the captured area. Next, the grid cells that correspond to the captured area become the “boundary” or “inner.” The remaining grid cells which are not in a captured area are then excluded.

Our system assigns a directional vector (V) to the boundary cells: up, down, left, right, right and up, right and down, left and up, and left and down. The first four vectors are assigned to the straight boundary chunks, and the last four diagonal directional vectors are used for the corner chunks. For the inner cells, the system labels each cell as “walkable” for any cell that could be walked by a user, or “obstacle” for any cell occupied by a real environment obstacle. In characterizing each cell as “walkable” or “obstacle,” we used the inside-outside test between each cell of the grid and each obstacle shape captured by a user. Both “walkable” and “obstacle” cells are assigned with an up vector. Figure 10 shows an example of a captured layout and its representation in the virtual grid.

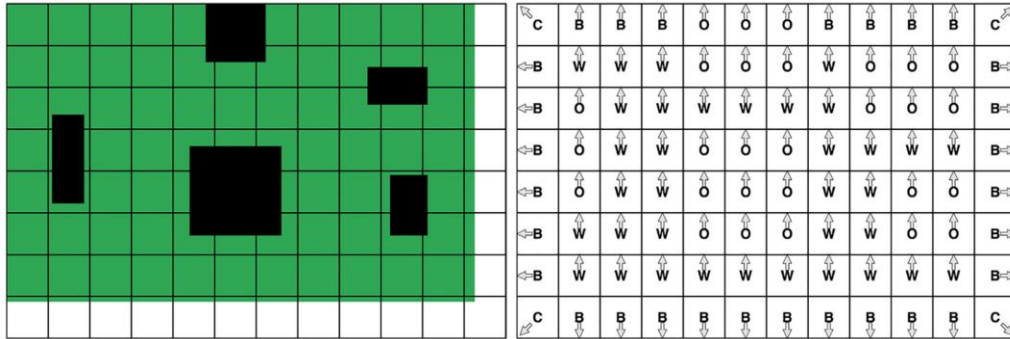


Figure 10: A layout of the real environment (left) and its representation based in the virtual grid (right). W stands for walkable, O stands for obstacle, and B stands for boundary grid cells. © [2021] Elsevier

2.4 Problem formulation and optimization

The goal of the proposed approach is to synthesize virtual reality game levels optimized for the real environment and its constraints and other design criteria, which are encoded by cost terms. Let $L = [c_{1,1}, \dots, c_{M,N}]$ denote the current configuration of the synthesized game level layout composed of several chunks $c_{i,j}$. Although the game levels chunks are represented in a 2D grid for simplicity, any chunk that belongs to the game level L will be represented as c_i . To

synthesize a game level, we developed a total cost function $C_{Total}(L)$ that evaluates the quality of a level L based on a number of costs (game level layout design decision):

$$C_{Total}(L) = w_M C_M(L) + w_F C_F(L) + w_V C_V(L) + w_A C_A(L) \quad (10)$$

C_M is the mapping term that attempts to map the game level chunks according to the input information represented as virtual grid array after the grid characterization process. C_F ensure a one-to-one fitting between the captured area and the synthesized game level. The C_V denotes the variation that could be introduced to the composed game level layout. The C_A represents the accessibility term that evaluates whether the user will be able to access any open space chunk in the synthesized level layout. Finally, the w_M , w_F , w_V , and w_A are weights that correspond to the cost terms and prioritize each cost term differently during the optimization process depending on their weighted value, given that $\{w_M, w_F, w_V, w_A\} \in [0, 1]$.

Various cost terms could be implemented to handle the layout synthesis of a game level. However, in this implementation phase, we limited the level layout cost terms to those most important for this project. The cost terms and the optimization process are presented in the subsections below.

2.4.1 Mapping cost

We implemented a level layout mapping cost that tries to map the chunks composing the game level based on the labeled grid of the real environment layout. To do so, we defined the following cost term:

$$C_M(L) = \frac{1}{|L|} \sum_{(c_i, g_i)} \Gamma_{(c_i, g_i)} \quad (11)$$

where c_i denotes a chunk of the game level, and g_i denotes a grid layout cell. (c_i, g_i) is then computed based on the following condition:

$$\Gamma_{(c_i, g_i)} = \begin{cases} 0 & \text{if } \mathcal{L}(c_i) = \mathcal{L}(g_i) \text{ and } \mathcal{V}(c_i) = \mathcal{V}(g_i) \\ 1 & \text{otherwise} \end{cases}$$

where $\mathcal{L}(\cdot)$ returns the label information of the chunk c_i and grid cell g_i , respectively. $\mathcal{V}(\cdot)$ returns the vector information of the chunks c_i and grid cell g_i , also respectively. This ensures that the $C_M(L)$ cost term returns a high value when there is an inconsistency between the layout of the synthesized game level and the target grid map information. Conversely, the cost term returns a low value when the synthesized game level is mapped correctly into the grid map information.

2.4.2 Fitting cost

To synthesize a game level layout that matches the layout of the real environment as close as possible, we introduced the use of the fitting cost function that attempts to minimize the difference between the area that the game level layout occupies and the area that is captured by a user using the virtual reality controller; this is our input area. This step is achieved by tweaking parts of the game level chunks (e.g., moving the fence of the boundary game level chunk closer to the real environment's captured boundary or selecting another boundary fence game object from our dataset that better fits the target grid). The fitting cost term is then represented as:

$$C_F(L) = \left(\frac{1}{\mathcal{N}_L} \mathcal{A}_B(L) - \underbrace{\frac{1}{\mathcal{N}_R} \mathcal{A}(R)}_{target} \right)^2 \quad (12)$$

where $\mathcal{A}_B(L)$ denotes the areas of the level layout L inside the boundaries and $\mathcal{A}(R)$ denotes the captured area of the real environment. Finally, \mathcal{N}_L and \mathcal{N}_R are normalization constants.

2.4.3 Variations cost

We realize that, as long as the information provided by the same real environment, the grid array remains the same (i.e., the topology and size of the real environment and the obstacles located in it do not change at all). Thus, the synthesized game levels would have minimal to no difference with one another; and therefore, the synthesized game level could be considered monotonic, and the game players might become bored quickly. In order to synthesize game levels that differ from one another and keep the players engaged as the game levels progress, a variation cost term was introduced to our total cost function. This cost term ensures that each generated game level will not look the same. For the level variation cost term, we use as input the grid array generated according to the real environment. We apply a Perlin noise [37] function to synthesize an intermediate game level layout I . Then, the variations cost is defined as:

$$C_V(L) = \left(\frac{1}{|L|} \sum_{c_i} |L(c_i) - I(c_i)| - \underbrace{\sigma_V}_{target} \right)^2 \quad (13)$$

where $\sigma_V \in [0, 1]$ is the target difference between the level layout L and the intermediate level layout I composed of c_i game level chunks. For example, $\sigma_V = 0.50$ means that the intermediate level layout I and the synthesized level layout L are 50% similar. Figure 11 illustrates

different intermediate level layouts placed on top of the level layout based on different variation targets (σ_V). It should be noted that with the proposed method the variation can be controlled by the user; therefore, the user can choose the amount of variation that will appear in the synthesized game level.

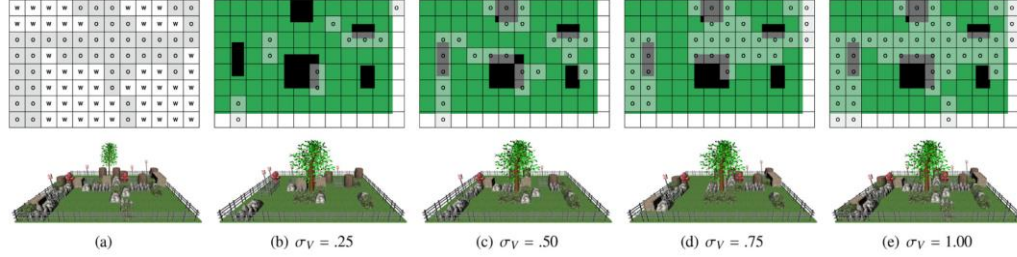


Figure 11: The layout of an intermediate game level synthesized using the Perlin noise (a), and the layout of the final synthesized game level layout in which the Perlin noise is included with different σ_V targets (4(b)–(e)). For all examples, the variation cost is given high priority, $w_V = 1.00$, and the accessibility cost is given low priority, $w_A = .05$. © [2021] Elsevier

2.4.4 Accessibility cost

Because of the variations cost, we understand that a synthesized game level might become over-occupied with obstacle chunks, blocking walkable areas that would otherwise be accessible to a user (see Figure 11(d) and (e)). To overcome this limitation, we included in our total cost function the accessibility cost that penalizes a synthesized game level when there are unoccupied grid cells that are not accessible. Our accessibility cost term is represented as:

$$C_A(L) = \frac{1}{|L|} \sum_{(c_i, c_j)} \Pi_{(c_i, c_j)} \quad (14)$$

which detects whether all open space area chunks c_i are accessible from any other open space chunk c_j . $\Pi_{(c_i, c_j)}$ is computed based on the following condition:

$$\Pi_{(c_i, c_j)} = \begin{cases} 0 & \text{if } \pi : c_i \rightarrow c_j \\ 1 & \text{otherwise} \end{cases}$$

In the condition above, π denotes the path between c_i and c_j . To compute whether a path exists between c_i and c_j , we used a simple pathfinding algorithm, the Depth-First Search [38][39]. Thus, if the pathfinding algorithm cannot connect two walkable area chunks, it returns 1 and forces the optimizer to continue searching for a game level layout by generating new intermediate-level layouts $I(c_i)$. Otherwise, if a path between c_i and c_j exists, the cost for that cell becomes 0; therefore, the optimizer can achieve its goals. It should be noted that there are cases in which

blocked areas might appear because of the capture process of the real environment and its constraints. Because this blocked area results from the real environment’s initial capture process and not due to the variation cost, our system does not consider that area as blocked because of the intermediate game level layout $I(c_i)$. Therefore, it does not penalize the $C_A(L)$ cost term.

2.4.5 Optimization

The game level layout L is optimized for the defined total cost function C_{Total} . A Markov Chain Monte Carlo optimization technique, known as simulated annealing [40] with a Metropolis-Hastings state searching step [41], was applied to solve this optimization problem. For this, we first define a Boltzmann-like [42] objective function:

$$f(L) = \exp\left(-\frac{1}{t}C_{Total}(L)\right) \quad (15)$$

where t denotes the temperature parameter of the simulated annealing. In each iteration of the optimization process, a move is applied to the current game level layout configuration L to propose a new configuration of the level layout L' . In the current implementation, the following moves were implemented:

- **Replace a chunk:** from the current game level layout configuration, a chunk c_i is randomly selected and replaced with another randomly chosen chunk from the chunks dataset.
- **Swap chunks:** from the current game level layout configuration, two chunks are randomly selected; the two chunks then swap positions.
- **Rotate a chunk:** our system randomly selects a c_i chunk and rotates it either to -90 or $+90$ degrees. This move helps the optimization to align the boundary chunks in the layout.
- **Edit a chunk:** our system randomly selects a boundary or obstacle chunk c_i and edits it by moving its child object (e.g., a fence) in either a positive or negative direction of the assigned vector.

The probability of selecting the move to be applied during the optimization process at each iteration was pre-defined by the authors. Unless otherwise specified, for the “replace a chunk” move, we set the selection probability to $p_{replace} = 0.30$. For the “swap chunks” move, we set

the selection probability to $p_{\text{swap}} = 0.20$. For the “rotate a chunk” move, we set the selection probability to $p_{\text{rotate}} = 0.20$, and for the “edit a chunk” move, we set the selection probability to $p_{\text{edit}} = 0.30$.

The output of each move is the proposition of a new configuration of the game level layout L . To decide whether the developed method should accept the proposed level design L' , the proposed total cost value $C_{\text{Total}}(L')$ is computed and compared to the cost of the current game level layout configuration $C_{\text{Total}}(L)$. To maintain the detailed balance in the optimization, our approach accepts the proposed level L with the acceptance probability $P(L'|L)$ based on the Metropolis criterion for each move as follows:

$$P(L'|L) = \min\left(1, \frac{f(L')}{f(L)}\right) \quad (16)$$

To efficiently explore the solution space, simulated annealing [40] was applied. Simulated annealing is controlled by a temperature parameter t that, at the beginning of the optimization, is assigned a high temperature t value to allow the optimizer to explore the synthesis of the game level solution space aggressively. During the optimization process, the temperature values t is lowered gradually. In the current scheme, we set $t = 1.00$ at the beginning of the optimization and decided to decrease it by 0.10 at every 1000 iterations until it reaches zero. Essentially, the optimizer becomes increasingly greedy in seeking to refine the solution while it is set to terminate if the total cost change is less than 5% over the previous 100 iterations. Based on our experimentation, we set the weights of each component of the total cost function in our optimization as $w_M = 1.00, w_F = 1.00, w_V = 0.50, w_A = 1.00$, unless otherwise specified.

By changing the weight of each cost term, the game level designer can always control the synthesis of the layout to emphasize specific design goals. We think that providing the game level designer the freedom to interactively explore possible game level layout designs can be a helpful feature. For example, the designer might want to prioritize the synthesized game level in a particular way by simply adjusting the weights on the total cost function (e.g., if a designer wants to assign a lower priority to the accessibility cost term, the weight of the accessibility cost term should be set be $w_A < 1.00$). The optimizer is responsible for generating the level layout and proposing a game level design for the designer-specified priorities. Figure 12 shows examples of game levels (the two games presented in Section 5) synthesized for different real environment layouts and different weights assigned to each component of the total cost function.

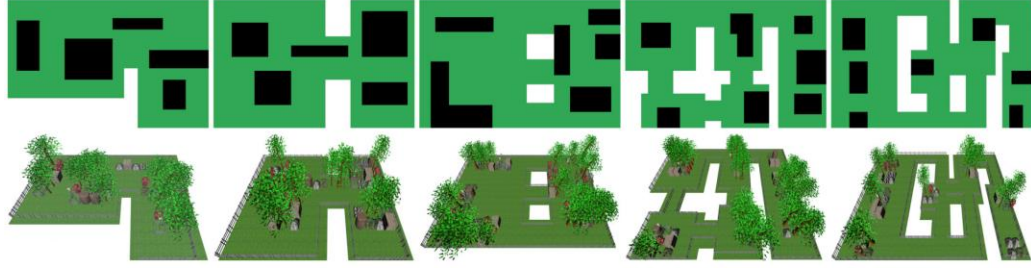


Figure 12: Example of synthesized level layouts based on different input layouts and weights assigned to each cost term of the total cost function. For all examples, the weight of the variation cost is set to $w_V = .00$. © [2021] Elsevier

2.5 Implementation details and example games

This section provides details about the implementation of our proposal and the two games that were developed.

2.5.1 Implementation

Our virtual reality game level design framework was implemented on a Dell Alienware with Intel a Core i7 CPU and 32 GB of memory. Our framework was developed in the Unity game engine using C#. Our scripts are easily adaptable to different game level chunks. The user simply needs to attach the necessary chunks to the Inspector window editor that has been implemented. After providing input about the layout of the real environment, our system automatically synthesizes the game level layout. The games presented in the subsections below were also implemented in the Unity game engine using the Oculus Integration. Our application was implemented and exported to Oculus Quest. Depending on the size of a real environment, synthesizing a game level consisting of 100 chunks (e.g., 10×10 grid) requires less than 5000 iterations. Based on our current implementation running on the Oculus Quest head-mounted display, this process can be finished in less than one minute. Finally, both games run on 65 fps in Oculus Quest. Moreover, we tested the number of iterations needed to synthesize game levels with different grid resolutions (5×5 , 10×10 , 15×15 , and 20×20 grids). In this evaluation for each grid resolution, we developed input layouts that are occupied with 0%, 25%, 50%, 75% and 100% obstacles and we ran the optimization process five times. The results are shown in Figure 13. As we can observe, the iterations needed are not related to the obstacle percentage but to the size of the grid.

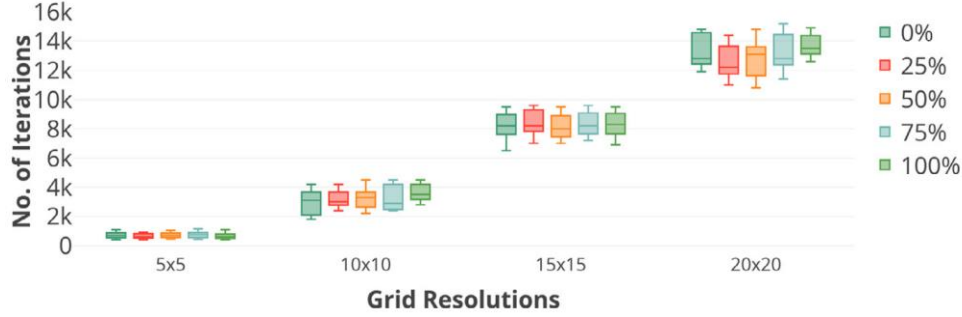


Figure 13: Number of iterations needed by our system to synthesize game level layouts with different grid resolutions (5×5 , 10×10 , 15×15 , and 20×20 grids) and different percentages (0%, 25%, 50%, 75% and 100%) of obstacles that occupy the grids. © [2021] Elsevier

A comparison of the virtual reality game level layout optimization between the MCMC and Greedy algorithms is shown in Figure 14. It should be noted that, compared to MCMC, the greedy algorithm only accepts a proposed total cost $C_{Total}(L')$ that provides a better configuration than the current total cost $C_{Total}(L)$. The MCMC algorithm obtains a solution with a lower total cost value (0.08) compared to the Greedy approach (0.47). The total cost value of the greedy algorithm experiment did not change from about iteration 550 to about iteration 650. Thus, the greedy optimization stopped at about iteration 650. Since the MCMC algorithm can accept a solution with a cost higher than that of the current solution with a certain acceptance probability, the sampling is capable of jumping out from a locally optimal solution. This prevents the sampling from being performed locally, and eventually locating a more optimal solution with a lower total cost value. Thus, the MCMC optimization stopped at about iteration 910.

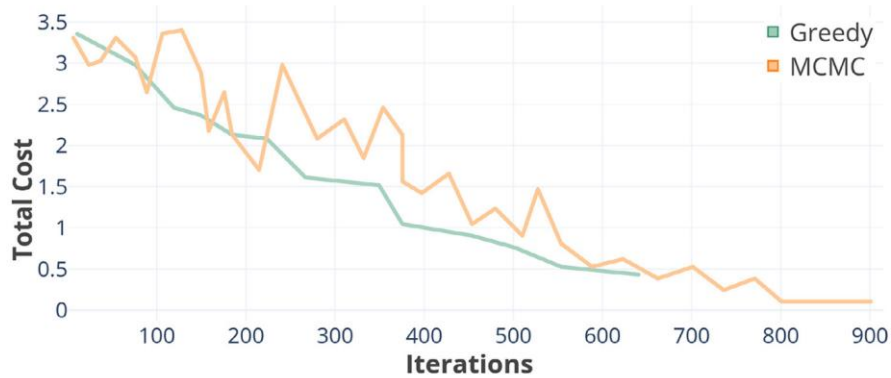


Figure 14: A comparison between optimizing the game level layout using the MCMC and the greedy algorithm in a 5×5 grid. The MCMC algorithm achieves lower minima compared to the greedy algorithm. © [2021] Elsevier

2.5.2 Example games

We developed two virtual reality games to demonstrate how our game level layout method can be used in gaming scenarios. The first game is called *Backyard Fortune*, a puzzle game. The second game is called *The Rebooter*, a shooting game. The Backyard Fortune game was the basis of the user study presented in Section 6. Below we provide more details about the games. Screenshots of the two games from the player’s point of view are shown in Figure 15. Moreover, Figure 16 shows a user playing the Backyard Fortune game in our lab space.

Backyard fortune The primary concept behind Backyard Fortune is for the player to collect puzzle pieces, find the key to a treasure box, and unlock it. The player is free to move in the walkable area of the game level and collect puzzle pieces using the Oculus Touch controller. Two panels on the virtual controllers are positioned where the knuckles would usually be placed. The first panel is designed like a clipboard, which provides instructions to the player on how to play the game. The second panel is an inventory that keeps track of which pieces have been collected in relation to the total number of puzzle pieces in the level. These panels can be toggled. The user can hide them and bring them back if he/she wants to keep track of how many more pieces need to be collected. The puzzle pieces are randomly placed in the walkable area once the level layout has been generated. The primary objective is to navigate the environment and use the controllers to collect the puzzle pieces and put them in their inventory. Once all the pieces have been collected, a key that unlocks the treasure box appears. The player can then pick up the key, insert it in the front of the treasure box, and unlock the box to enjoy his/her fortune.

The rebooter The Rebooter game presents more challenging conditions to the player. Specifically, the player tries to shoot the enemies that chase him/her. The enemies have been designed so that they have a set patrol path around the obstacle piece. The set patrol path is included to satisfy one of the core gameplay mechanics of video games—anticipating and reacting to non-playable characters’ patterns. In this game, the player holds a virtual gun in his/her right hand that shoots lasers. The laser gun is used to fulfill the instruction to destroy any enemy. The game ends once the player destroys all enemies.

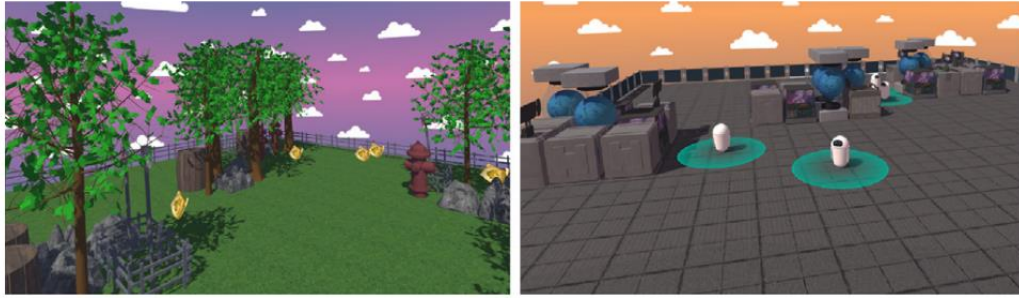


Figure 15: Screenshots of the Backyard Fortune (left) and The Rebooter (right) games.
© [2021] Elsevier



Figure 16: A user playing our Backyard Fortune game in our lab space. © [2021] Elsevier

2.6 User study

A user study was conducted to evaluate the proposed optimization-based game level design method. The following sections explain the details of our study.

2.6.1 Participants

The recruitment of participants was based on emails that were sent to all students in our department. In total, 25 students volunteered for the study (18 males and 7 females). Participants ranged in age from 19 to 27, with the mean age $M = 21.56$ ($SD = 2.78$). All participants had prior experience with virtual reality games. There was no compensation for participating in the study. The study followed a within-group design; therefore, all participants experienced all three conditions presented in the section below.

2.6.2 Experimental conditions

Three experimental conditions were developed to evaluate participants' experiences: (1) when interacting with our method in which the virtual environment is optimized based on the real environment and its constraints; (2) when interacting with a real environment that mismatches the constraints of the virtual environment; and (3) when interacting with a free-from-obstacles real environment. For this study, the Oculus Quest head-mounted display was used. Note that the synthesized game level was the same for all three conditions for all participants, which means that the game level layout that was optimized in the first condition was also used for the rest of the conditions. The only difference was in the layout of the real environment and the manner in which the users were made aware of the real environment obstacles. Details of the three conditions are provided below.

- **Optimization:** For this condition, the layout of the real environment and the obstacles located in it were captured using our method. Then, the game level's layout was automatically generated based on the proposed method. In doing so, the boundary game level chunks were placed on the boundaries (walls) of the real environment. Virtual obstacles were placed as substitutes in the exact position of real environment obstacles.
- **No Optimization:** This condition was used to determine whether mismatching the real and virtual environments in terms of obstacles placed in the real environment would affect participants' responses. For this condition, we are using the real environment and its constraints that were initially captured using our method. Then, the real environment obstacles (carton boxes) were moved to a different position.

Finally, the real environment is captured using the calibration tool in Oculus Quest; therefore, a mismatching between the real and the virtual environment was achieved. During the gameplay, the user is informed if he/she is close to a real environment obstacle by the guardian functionality of Oculus Quest.

- **No Obstacles:** This is considered our baseline condition and was used to investigate how participants interact in the real environment while knowing in advance that there are no obstacles (the carton boxes were removed from the play area); this kind of real environment could be considered as safe. As in the other two conditions, we are using the game level layout that was generated by the optimization condition. Boundary game level chunks were placed in the virtual environment to inform the participant about the actual boundaries of the real environment and obstacle chunks were placed at the initial positions of real environment obstacle.

2.6.3 Measurements

In this study, a computer-based questionnaire was provided to all participants. The purpose was to explore their presence and fear of collision (emotional state) with the virtual environment. Specifically, the sense of presence was measured using the Igroup Presence Questionnaire (IPQ) [43][44], which consisted of 14 items and was divided into four parts: (1) one item reflected the initial definition of **presence**, according to Slater and Usoh [45]; (2) five items reflected **spatial presence**, denoting the sense of being “physically there” in the virtual environment; (3) four items reflected **involvement** focused on attention during the interaction, as well as the perceived involvement of the participants; and (4) four items reflected the **experienced realism**, which evaluates the realism of the virtual environment. The four-item scale on emotion, drawn from Tcha-Tokey et al. [46], was also used to investigate participants’ **fear** (emotional state) while in the virtual environment. Finally, a section of the questionnaire asks participants for additional input about their experience when interacting with the three experimental conditions outlined in Section 6.2.

2.6.4 Procedure

When participants entered the lab, the research team asked them to sign a consent form approved by the Institutional Review Board of our university if they agree to participate in the study. Then, the participants were then asked to complete a demographics questionnaire.

As mentioned, Oculus Quest does not allow passthrough access; therefore, a member of the research team that was experienced with the capturing process used the Oculus Quest to capture the real environment and its constraints, and during this process the researcher took slightly off the head-mounted display to observe the real environment. The participants were not given any information about the conditions they would experience. They would first see the virtual environment once they put on the head-mounted display, and then the game would start. The research team helped the participants with the virtual reality equipment (Oculus Quest) before the game started. Once the virtual reality gaming application started, the participants were asked to play the game. When the game was over, a visual indication on the screen would notify the participant. The research team then helped the participants by setting up the next experimental condition.

To control potential carry-over effects, the sequence in which each participant would experience the three experimental conditions was randomized using Graeco-Latin squares. Between the conditions, the participants were asked to complete a questionnaire distributed in a paper-based format. This time period was also used to provide participants with a short break. The participants were informed that the virtual environment's boundaries corresponded to the boundaries of the real environment. However, participants were not told whether there was a match or mismatch between the real and the virtual environments. They were able to observe it once they put on the head-mounted display. It should be noted that none of the participants made contact with any of the walls during the study. Each participant spent no more than 45 minutes completing the study. All participants were aware that they were free to quit the study at any time.

2.6.5 Results

In analyzing our data, we used a one-way repeated measures analysis of variance (ANOVA) to determine the differences across the three experimental conditions. The internal validity of the scales of the questionnaire was measured using Cronbach's alpha coefficient. With sufficient

scores ($0.73 < \alpha < 0.94$), we used a cumulative score for each item that belonged to each questionnaire component. The normality assumption of the objective measurements and subjective ratings were evaluated with Shapiro–Wilk tests at the 5% level and with the residuals' graphic Q-Q plots. Post hoc comparisons were conducted using Bonferroni corrected estimates. A $p < 0.05$ value was deemed statistically significant. Boxplots from the obtained results are shown in Figure 17.

Statistically significant results were found for **presence** across the examined experimental conditions [$\Lambda = 0.603, F(2, 23) = 7.560, p < 0.005, \eta_p^2 = 0.397$]. The post hoc comparison showed that the mean score of the no optimization condition ($M = 3.24, SD = 1.56$) was lower than that of the no obstacle condition ($M = 4.88, SD = 1.42$) at the $p < 0.005$ level and the optimization condition ($M = 4.64, SD = 1.60$) at the $p < 0.05$ level.

The **spatial presence** of participants was statistically significant across the examined experimental conditions [$\Lambda = 0.504, F(2, 23) = 11.220, p < 0.001, \eta_p^2 = 0.494$]. The post hoc comparison showed that the mean score of the no optimization condition ($M = 3.52, SD = 1.61$) was lower than that of the no obstacle condition ($M = 5.56, SD = 1.38$) at the $p < 0.005$ level and the optimization condition ($M = 4.92, SD = 1.52$) at the $p < 0.001$ level.

We also identified a statistically significant effect on the participants' **involvement** across the examined experimental conditions [$\Lambda = 0.613, F(2, 23) = 7.273, p < 0.005, \eta_p^2 = 0.387$]. The post hoc comparison showed that the mean score of the no optimization condition ($M = 3.36, SD = 1.80$) was lower than that for the no obstacle condition ($M = 4.96, SD = 1.59$) at the $p < 0.01$ level and the optimization condition ($M = 4.56, SD = 1.44$) at the $p < 0.05$ level.

Notably, no statistically significant results were found for the **experienced realism** measurement across the examined experimental conditions [$\Lambda = 0.930, F(2, 23) = .864, p = 0.435, \eta_p^2 = 0.070$]. However, a statistically significant effect on the participants' **fear** (emotion) was found across the examined experimental conditions [$\Lambda = 0.328, F(2, 23) = 23.593, p < 0.001, \eta_p^2 = 0.672$]. The post hoc comparison showed that the mean score of the no obstacle condition ($M = 2.32, SD = 1.10$) was lower than that for the no optimization condition ($M = 4.88, SD = 1.53$) at the $p < 0.001$ level and the optimization condition ($M = 3.48, SD =$

1.53) at the $p < 0.05$ level. Moreover, we found that the mean fear rating of the optimization condition was lower than the no optimization condition at the $p < 0.05$ level.

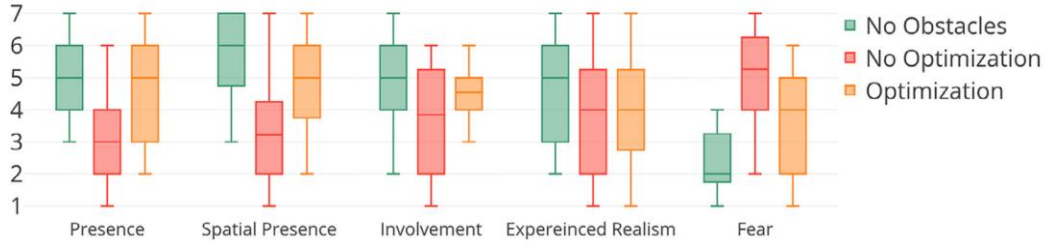


Figure 17: Boxplots from self-reported ratings for each examined concept across the three experimental conditions. © [2021] Elsevier

2.6.6 Discussion

The analyses of presence, spatial presence, and involvement revealed that our method could in fact synthesize the virtual reality game level based on the real environment and its constraints and further synthesize game levels that keep the user engaged. Specifically, our optimization-based method (optimization condition) was able to outperform the no optimization condition. Please note that the no optimization condition describes the way that people experience virtual reality games from their living room; in other words, from a real environment full of obstacles that do not match the virtual environment in terms of constraints and appearance. Additionally, it appeared that, during the gaming experience, the presence of our participants was not interrupted (break-in-presence effect [5]) by the guardian functionality of Oculus. As a result, our participants were more able to focus on and enjoy the game during the optimization and no obstacle conditions.

Moreover, the statistical analyses showed that the proposed method could indeed provide results close to the no obstacle condition. This encouraging finding means that the participant level of presence, even in a constrained environment, was close to the level of presence in a free-from-obstacles environment. This finding indicates that when participants are placed in a virtual environment that matches the constraints of the real environment, their presence level is close to a condition during which they know in advance that they will be able to move and interact in a free-from-obstacles environment, even if there is no appearance matching between the two environments. Finally, for experienced realism, participants experienced the same virtual

environment across the three conditions. As a result, they provided similar ratings since the experienced realism was related more to the appearance of the virtual environment and less on the structure of the real environment [47][48].

In addition to the positive results regarding presence, we also observed interesting findings relating to the participants' emotional state when examining their fear during their interaction with the three different experimental conditions. Specifically, participants indicated that when playing the game during the no obstacle condition, their level of fear was lower than when playing in the no optimization and the proposed optimization conditions. This result indicates that when participants were placed in a free-from-obstacles real environment, they felt safer walking in it because simply there were no obstacles to anticipate or avoid. However, once obstacles were placed, the participants began to feel less safe since they needed to move more carefully in order to avoid any potential sudden encounters like hitting an obstacle. Moreover, our results showed that participants' fear was rated lower during the optimization condition than in the no optimization condition. The finding related to fear is significant because it demonstrates that, to the extent participants become aware that there is a match between the real and the virtual environment in terms of layout and constraints, this spatial awareness reduced the fear of colliding with the obstacles. We consider this to be the most important finding of the study that highlights the advantage of the proposed approach.

The participants also submitted several comments about the different conditions they experienced. All of the comments we received for the optimization-based approach were positive, suggesting that optimizing virtual reality game levels for real environments and their constraints should be taken into consideration by virtual reality game developers. For the optimization condition, some of the participants reported that once they became familiarized with the virtual environment, they realized there was a match between the position of the real environment obstacles and the virtual objects they were viewing. A few more said that the game objects in the virtual environment made the location of the walkable area clear and therefore easy for them to navigate freely in the virtual environment. Regarding the no optimization condition, some participants reported they disliked the interruption of the Oculus' guardian functionality. Moreover, others noted that such mismatching between the real and the virtual environment made them more apprehensive before performing their next step.

In conclusion, we realized it would have been useful to collect additional data to further understand how participants interacted in the virtual environment during the three conditions. In particular, added feedback on data such as participants' movements, proximity to boundaries, proximity to real and virtual obstacles, and the number of collisions with real obstacles would have provided additional and, potentially, useful information on the way they perceived and interacted between the real and virtual environments. However, considering the findings from the self-reported ratings and the comments participants submitted, we concluded that the proposed optimization method could be a promising solution for synthesizing virtual reality game levels for real environment constraints while also helping keep the user engaged with the game.

2.7 Conclusions, limitations, and future work

This paper introduced an optimization-based method of designing game levels based on real environment layouts and the constraints that vary in shape and size. We think that our method could be effectively used in a variety of real environments, including living rooms, bedrooms, or office spaces. The proposed method synthesizes game level layouts in a fast and scalable manner with minimum effort by the user and without the need for additional hardware (e.g., a 3D scanner) for providing input information. Additionally, our method provides game level designers with the necessary control of the synthesized game level in order to prioritize the objectives of the design process by simply changing the weight that controls the cost terms.

To understand the effectiveness of the proposed method, we conducted a user study. Our study provided a number of interesting insights into the participants' experience in the virtual environment. The results of the user study indicated that the proposed method was admittedly able to enhance the participants' presence and involvement while reducing the fear of collision with the real environment and its obstacles.

There are several limitations that should be addressed in the future. Tackling these limitations would allow modifications to the proposed method by making it applicable to a broader range of users and by enhancing its efficiency to accommodate more complex gaming scenarios. The proposed method examines only a small number of chunk types and chunks with simple shapes. However, there may be games that require a variety of complex chunks in terms of shape and size. Further, the proposed method looks at only the (x, z) coordinates of the real environment and its constraints, thereby excluding the height of the environment and the obstacles contained in

it. We think that incorporating the third dimension, or even a more precise representation of the real environment using a 3D scanner, would allow us to synthesize game levels that match the real environment more precisely in terms of shape and constraints.

In addition to the above-mentioned limitations, several alternative directions could also be explored in the near future. Specifically, the current method is highly dependent on the actual size of the real environment. We think that the implementation of a layer-based method similar to the Flexible Spaces [49] and the Impossible Spaces [50], in addition to our optimization-based approach, would allow game developers to design longer and more complex game levels (e.g., a dungeon-related game). Moreover, instead of using a small number of game level chunks, we think that experimentation with a large 3D dataset might be useful to generate game levels with enhanced appearance alternatives within the synthesized layout. We hope that more optimization-based approaches that synthesize virtual reality game levels for real environment constraints will be proposed in the near future.

2.8 Declaration of Competing Interest

Authors declare that they have no conflict of interest.

2.9 References

- [1] LaViola Jr, Joseph. J., Kruijff, Ernst., McMahan, Ryan. P., Bowman, Doug. and Poupyrev, Ivan. P. 2017. *3D user interfaces: theory and practice*. Addison-Wesley Professional. Retrieved
- [2] Slater, Mel. (2009). Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1535), 3549-3557.
- [3] Mousas, Christos., Kao, Dominic., Koiliias, Alexandros. and Rekabdar, Banafsheh. 2020. Real and virtual environment mismatching induces arousal and alters movement behavior. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (pp. 626-635). IEEE.
- [4] Slater, Mel. 2002. Presence and the sixth sense. *Presence*, 11(4), 435-439.
- [5] Slater, Mel., Brogni, Andrea. and Steed, Anthony. 2003. Physiological responses to breaks in presence: A pilot study. In *Presence 2003: The 6th annual international workshop on presence* (Vol. 157). Cleveland, OH: Citeseer.

- [6] Usoh, Martin., Arthur, Kevin., Whitton, Mary. C., Bastos, Rui., Steed, Anthony., Slater, Mel. and Brooks Jr, Frederick. P. 1999. Walking> walking-in-place> flying, in virtual environments. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques* (pp. 359-364).
- [7] Jonathan, Roberts, C., Razzaque, S., Kohn, Z. and Whitton, M. 2001. Redirected Walking. *Proceedings of Eurographics*. (pp. 1-6).
- [8] Suma, Evan. A., Bruder, Gerd., Steinicke, Frank., Krum, David. M. and Bolas, Mark. 2012. A taxonomy for deploying redirection techniques in immersive virtual environments. In *2012 IEEE Virtual Reality Workshops (VRW)* (pp. 43-46). Costa Mesa, CA, USA. doi: 10.1109/VR.2012.6180877.
- [9] Sun, Qi., Patney, Anjul., Wei, Li-yi., Shapira, Omer., Lu, Jingwan., Asente, Paul., Zhu, Suwen., McGuire, Morgan., Luebke, David. and Kaufman, Arie. 2018. Towards virtual reality infinite walking: dynamic saccadic redirection. *ACM Transactions on Graphics (TOG)*, 37(4), 1-13.
- [10] Sun, Qi., Wei, Li-Yi. and Kaufman, Arie. 2016. Mapping virtual and physical reality. *ACM Transactions on Graphics (TOG)*, 35(4), 1-12.
- [11] Nescher, Thomas., Zank, Markus. and Kunz, Andreas. 2016. Simultaneous mapping and redirected walking for ad hoc free walking in virtual environments. In *2016 IEEE Virtual Reality (VR)* (pp. 239-240). Greenville, SC, USA. doi: 10.1109/VR.2016.7504742.
- [12] Gal, Ran., Shapira, Lior., Ofek, Eyal. and Kohli, Pushmeet. 2014. FLARE: Fast layout for augmented reality applications. In *2014 IEEE international symposium on mixed and augmented reality (ISMAR)* (pp. 207-212). IEEE.
- [13] Nuernberger, Benjamin., Ofek, Eyal., Benko, Hrvoje. and Wilson, Andrew. D. 2016. Saptoreality: Aligning augmented reality to the real world. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (pp. 1233-1244).
- [14] McGill, Mark., Boland, Daniel., Murray-Smith, Roderick. and Brewster, Stephen. 2015. A dose of reality: Overcoming usability challenges in VR head-mounted displays. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (pp. 2143-2152).
- [15] Keller, Marilyn. and Exposito, Frederic. 2018. Game Room Map Integration in Virtual Environments for Free Walking. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (pp. 763-764). Tuebingen/Reutlingen, Germany. doi: 10.1109/VR.2018.8446385.
- [16] Keller, Marilyn. and Tchilinguirian, Tristan. 2019. Obstacles Awareness Methods from Occupancy Map for Free Walking in VR. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (pp. 1012-1013). Osaka, Japan: IEEE. doi: 10.1109/VR.2019.8798260.

- [17] Sousa, Mauricio., Mendes, Daniel. and Jorge, Joaquim. 2019. Safe walking in VR using augmented virtuality. arXiv preprint arXiv:1911.13032.
- [18] Fei. Wu and Evan. S. Rosenberg, 2019. "Combining Dynamic Field of View Modification with Physical Obstacle Avoidance," In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, Osaka, Japan, pp. 1882-1883, doi: 10.1109/VR.2019.8798015.
- [19] Kanamori, Kohei., Sakata, Nobuchika., Tominaga, Tomu., Hijikata, Yoshinori., Harada, Kensuke. and Kiyokawa, Kiyoshi. 2018. Obstacle Avoidance Method in Real Space for Virtual Reality Immersion. In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (pp. 80-89). Munich, Germany: IEEE. doi: 10.1109/ISMAR.2018.00033.
- [20] Sra, Misha., Garrido-Jurado, Sergio., Schmandt, Chris. and Maes, Pattie. 2016. Procedurally generated virtual reality from 3D reconstructed physical space. In *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology* (pp. 191-200).
- [21] Müller, Pascal., Wonka, Peter., Haegler, Simon., Ulmer, Andreas. and Van Gool, Luc. 2006. Procedural modeling of buildings. In *ACM SIGGRAPH 2006 Papers* (pp. 614-623).
- [22] Parish, Yoav. I. and Müller, Pascal. 2001. Procedural modeling of cities. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (pp. 301-308).
- [23] Smelik, Ruben. M., Tutenel, Tim., Bidarra, Rafael. and Benes, Bedrich. 2014. A survey on procedural modelling for virtual worlds. In *Computer Graphics Forum* (Vol. 33, No. 6, pp. 31-50).
- [24] Peng, Chi-Han., Yang, Yong-Liang. and Wonka, Peter. 2014. Computing layouts with deformable templates. *ACM Transactions on Graphics (TOG)*, 33(4), 1-11.
- [25] Jiang, Haiyong., Yan, Dong-Ming., Dong, Weiming., Wu, Fuzhang., Nan, Liangliang. and Zhang, Xiaopeng. (2016). Symmetrization of facade layouts. *Graphical Models*, 85, 11-21.
- [26] Ma, Chongyang., Vining, Nicholas., Lefebvre, Sylvain. and Sheffer, Alla. 2014. Game level layout from design specification. In *Computer Graphics Forum* (Vol. 33, No. 2, pp. 95-104).
- [27] Hartsook, Ken., Zook, Alexander., Das, Sauvik. and Riedl, Mark. O. 2011. Toward supporting stories with procedurally generated game worlds. In *2011 IEEE Conference on Computational Intelligence and Games (CIG'11)* (pp. 297-304). IEEE.
- [28] Hullett, Kenneth. and Mateas, Michael. 2009. Scenario generation for emergency rescue training games. In *Proceedings of the 4th International Conference on Foundations of Digital Games* (pp. 99-106).
- [29] Xie, Biao., Zhang, Yongqi., Huang, Haikun., Ogawa, Elisa., You, Tongjian. and Yu, Lap-Fai. 2018. Exercise Intensity-Driven Level Design. *IEEE Transactions on Visualization and Computer Graphics*, 24(4), 1661-1670. doi: 10.1109/TVCG.2018.2793618.

- [30] Zhang, Yongqi., Xie, Biao., Huang, Haikun., Ogawa, Elisa., You, Tongjian. and Yu, Lap-Fai. 2019. Pose-guided level design. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (pp. 1-12).
- [31] Liu, Huimin., Wang, Zhiqun., Mousas, Christos. and Kao, Dominic. 2020. Virtual reality racket sports: Virtual drills for exercise and training. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (pp. 566-576). IEEE.
- [32] Harrison, Brent. and Roberts, David. 2014. Analytics-driven dynamic game adaption for player retention in a 2-dimensional adventure game. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (Vol. 10, No. 1, pp. 23-29).
- [33] Luo, Linbo., Yin, Haiyan., Cai, Wentong., Lees, Michael. and Zhou, Suiping. 2013. Interactive scenario generation for mission-based virtual training. *Computer Animation and Virtual Worlds*, 24(3-4), 345-354.
- [34] Rowe, Jonathan., Mott, Bradford. and Lester, James. 2014. Optimizing player experience in interactive narrative planning: a modular reinforcement learning approach. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (Vol. 10, No. 1, pp. 160-166).
- [35] Smith, Adam. M., Andersen, Erik., Mateas, Michael. and Popović, Zoran. 2012. A case study of expressively constrainable level design automation tools for a puzzle game. In *Proceedings of the International Conference on the Foundations of Digital Games* (pp. 156-163).
- [36] Adams, Bart. and Dutré, Philip. 2003. Interactive boolean operations on surfel-bounded solids. In *ACM SIGGRAPH 2003 Papers* (pp. 651-656).
- [37] Perlin, Ken. 1985. An image synthesizer. *ACM Siggraph Computer Graphics*, 19(3), 287-296.
- [38] Cormen, Thomas. H., Leiserson, Charles. E., Rivest, Ronald. L. and Stein, Clifford. 2022. *Introduction to algorithms*. MIT press.
- [39] Tarjan, Robert. (1972). Depth-first search and linear graph algorithms. *SIAM journal on computing*, 1(2), 146-160.
- [40] Kirkpatrick, S., Gelatt Jr, C. D. and Vecchi, M. P. 1983. Optimization by simulated annealing. *science*, 220(4598), 671-680.
- [41] Chib, Siddhartha. and Greenberg, Edward. 1995. Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4), 327-335.
- [42] Cercignani, Carlo. 1988. "The boltzmann equation." pp. 40-103 in *The Boltzmann Equation and Its Applications*. Springer New York.

- [43] Regenbrecht, Holger. and Schubert, Thomas. 2002. Real and illusory interactions enhance presence in virtual environments. *Presence: Teleoperators & Virtual Environments*, 11(4), 425-434.
- [44] Schubert, T. W. 2003. The sense of presence in virtual environments: A three-component scale measuring spatial presence, involvement, and realness. *Z Medienpsychologie*, 15(2), pp. 69-71.
- [45] Slater, Mel., Usoh, Martin. and Steed, Anthony. 1994. Depth of presence in virtual environments. *Presence: Teleoperators & Virtual Environments*, 3(2), 130-144.
- [46] Tcha-Tokey, Katy., Christmann, Oliver., Loup-Escande, Emilie. and Richir, Simon. 2016. Proposition and validation of a questionnaire to measure the user experience in immersive virtual environments. *International Journal of Virtual Reality*, 16(1), 33-48.
- [47] Arthur, E., Hancock, P. A. and Chrysler, S. T. 1997. The perception of spatial layout in real and virtual worlds. *Ergonomics*, 40(1), 69-77.
- [48] Kuliga, S. F., Thrash, T., Dalton, R. C. and Hölscher, C. 2015. Virtual reality as an empirical research tool-Exploring user experience in a real building and a corresponding virtual model. *Computers, environment and urban systems*, vol. 54, pp: 363-375.
- [49] Vasylevska, Khrystyna., Kaufmann, Hannes., Bolas, Mark. and Suma, Evan. A. 2013. Flexible spaces: Dynamic layout generation for infinite walking in virtual environments. In *2013 IEEE Symposium on 3D User Interfaces (3DUI)* (pp. 39-42). IEEE.
- [50] Suma, Evan. A., Lipps, Zachary., Finkelstein, Samantha., Krum, David. M. and Bolas, Mark. 2012. Impossible spaces: Maximizing natural walking in virtual environments with self-overlapping architecture. *IEEE Transactions on Visualization and Computer Graphics*, 18(4), 555-564.

CHAPTER 3. PUBLISHED ARTICLE #3: SYNTHESIZING GAME LEVELS FOR COLLABORATIVE GAMEPLAY IN A SHARED VIRTUAL ENVIRONMENT

All the authors in the paper agreed on the author of the dissertation to use this publication in her Ph.D. dissertation.

This paper was published in ACM journal---Transactions on Interactive Intelligent Systems (TiiS).

© [2023] ACM. Reprinted, with permission, from [Liu, Huimin., Choi, Minsoo., Kao, Dominic. and Mousas, Christos, Synthesizing Game Levels for Collaborative Gameplay in a Shared Virtual Environment. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 2023]

(Cite) Liu, H., Choi, M., Kao, D., & Mousas, C. (2023). Synthesizing Game Levels for Collaborative Gameplay in a Shared Virtual Environment. *ACM Transactions on Interactive Intelligent Systems (TiiS)*. <https://doi.org/10.1145/3558773>

Copyright © 2023 Copyright held by the owner/author(s).

The authors confirm contribution to the paper as follows:

- Study Conceptualization and Design: Mousas, Christos and Kao, Dominic;
- Methodology and/or Simulation Design: Liu, Huimin and Choi, Minsoo;
- Execution/Implementation: Liu, Huimin and Choi, Minsoo;
- Data collection: Liu, Huimin and Choi, Minsoo;
- Analysis and interpretation of results: Mousas, Christos, Choi, Minsoo, and Liu, Huimin;
- Findings, Conclusions, and/or Recommendations: Mousas, Christos and Kao, Dominic;
- Draft manuscript preparation: Mousas, Christos, Choi, Minsoo, and Liu, Huimin.

All authors reviewed the results and approved the final version of the manuscript.

All authors in this paper agree on the usage of the paper in this dissertation.

Purdue does not require any formal permission.

Abstract

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 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.

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

3.1 Introduction

In our daily lives, we collaborate with others on various tasks in various ways. According to Webster’s Dictionary, “collaborations”⁸ 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

⁸ <https://www.merriam-webster.com/thesaurus/collaboration>.

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*⁹) 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 (Figure 18). 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 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 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

⁹ https://www.mariowiki.com/Super_Mario_Land.

designer-defined parameters in a few seconds, 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 3.2, we present related work on collaborative games and virtual reality experiences. In Section 3.3, we describe the preliminary remarks of our project. In Section 3.4, we explain the formulation of the game level synthesis and the optimization process. In Section 3.5, we outline the conducted user study and discuss our findings. In Section 3.6, we review the limitations of our method. Finally, in Section 3.7, we present our conclusions and potential future research directions.



Figure 18: Our method synthesizes a game level in which participants collaborate in a shared virtual environment to play a game.

3.2 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*,¹⁰ *Trine*,¹¹ and *Keep Talking and Nobody Explodes*.¹² 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*

¹⁰ <https://www.thinkwithportals.com/>.

¹¹ <https://www.frozenbyte.com/games/>.

¹² <https://keeptalkinggame.com/>.

*Legends*¹³ 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*,¹⁴ allow many players to collaborate in various tasks, such as fighting a dragon. According to Wikipedia's list of cooperative video games,¹⁵ some MMORPGs can be played by players ranging from two, such as *Space Duel*¹⁶ and *Sky Force*,¹⁷ to 128, such as *Freelancer*¹⁸ and *The Forest*.¹⁹

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,"

¹³ https://en.wikipedia.org/wiki/League_of_Legends.

¹⁴ https://en.wikipedia.org/wiki/World_of_Warcraft.

¹⁵ https://en.wikipedia.org/wiki/List_of_cooperative_video_games.

¹⁶ https://en.wikipedia.org/wiki/Space_Duel.

¹⁷ https://en.wikipedia.org/wiki/Sky_Force.

¹⁸ [https://en.wikipedia.org/wiki/Freelancer_\(video_game\)](https://en.wikipedia.org/wiki/Freelancer_(video_game)).

¹⁹ [https://en.wikipedia.org/wiki/The_Forest_\(video_game\)](https://en.wikipedia.org/wiki/The_Forest_(video_game)).

“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, Luaret²⁰ 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, Redding²¹ 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) 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*²² game

²⁰ https://www.gamasutra.com/view/news/328756/The_four_atoms_of_cooperative_video_games.php.

²¹ <https://www.gdcvault.com/play/1014379/Keep-it-Together-Encouraging-Cooperative>.

²² https://en.wikipedia.org/wiki/Jamestown:_Legend_of_the_Lost_Colony.

provided practical guidelines on designing collaborative games based on player observations²³ 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*²⁴ game followed similar rules to establish a relationship between the players.²⁵ 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

²³ <https://www.co-optimus.com/editorial/976/page/1/indie-ana-co-op-and-the-dev-stories-you-re-all-in-this-together.html>.

²⁴ <https://togetherthegame.com/>.

²⁵ <https://www.co-optimus.com/editorial/1376/page/1/indie-ana-co-op-and-the-dev-stories-fostering-gaming-relationships.html>.

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 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 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*²⁶ and *Elite*²⁷, 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 it 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

²⁶ [https://en.wikipedia.org/wiki/Rogue_\(video_game\)](https://en.wikipedia.org/wiki/Rogue_(video_game)).

²⁷ [https://en.wikipedia.org/wiki/Elite_\(video_game\)](https://en.wikipedia.org/wiki/Elite_(video_game)).

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.

3.3 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.



Figure 19: 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.

3.3.1 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. Figure 19 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], Luaret²⁰, and Redding²¹, 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.

Figure 20 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.

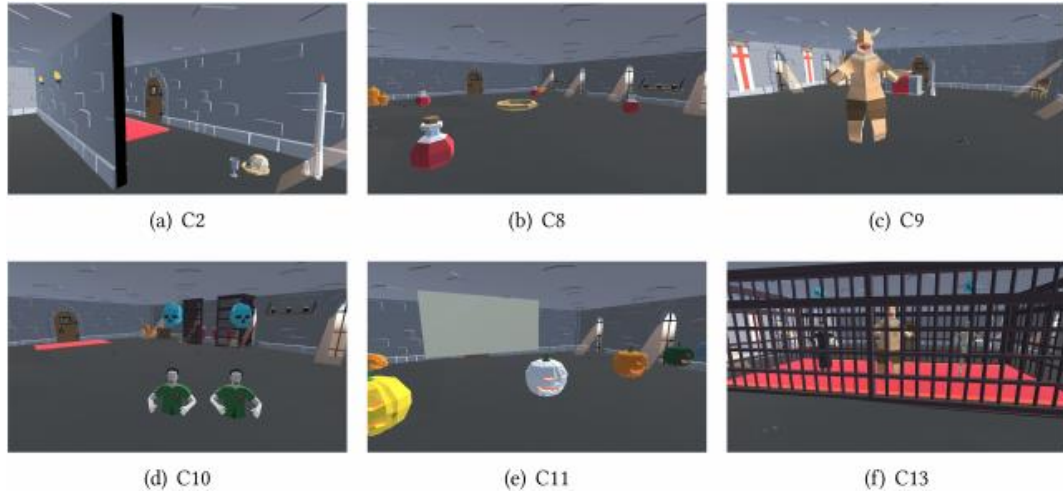


Figure 20: Example scenes of the developed game level chunks from a first-person perspective.

3.3.2 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. Figure 19 illustrates the collaboration zones of different game level chunks.

For example, in the case of the C6 game level chunk (Figure 19(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 (see APPENDIX; Figure 30-44) 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.

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	***

3.4 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.

3.4.1 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:

$$C_{\text{Total}}(L) = \mathbf{w}_{\text{Collab}}^T \mathbf{C}_{\text{Collab}} + \mathbf{w}_{\text{Prior}}^T \mathbf{C}_{\text{Prior}} \quad (17)$$

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 as the size of the game level (number of game level chunks) and repetition among adjacent game level chunks. As mentioned before, $\mathbf{w}_{\text{Prior}} = [w_{\text{Prior}}^S, w_{\text{Prior}}^R]$ 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.

3.4.2 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:

$$C_{Collab}^M(L) = \left(\frac{1}{|L|} \sum_{c_i} \mathcal{D}(c_i) - \rho_M \right)^2 \quad (18)$$

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. Figure 21 illustrates the game levels synthesized by varying the value of ρ_M .

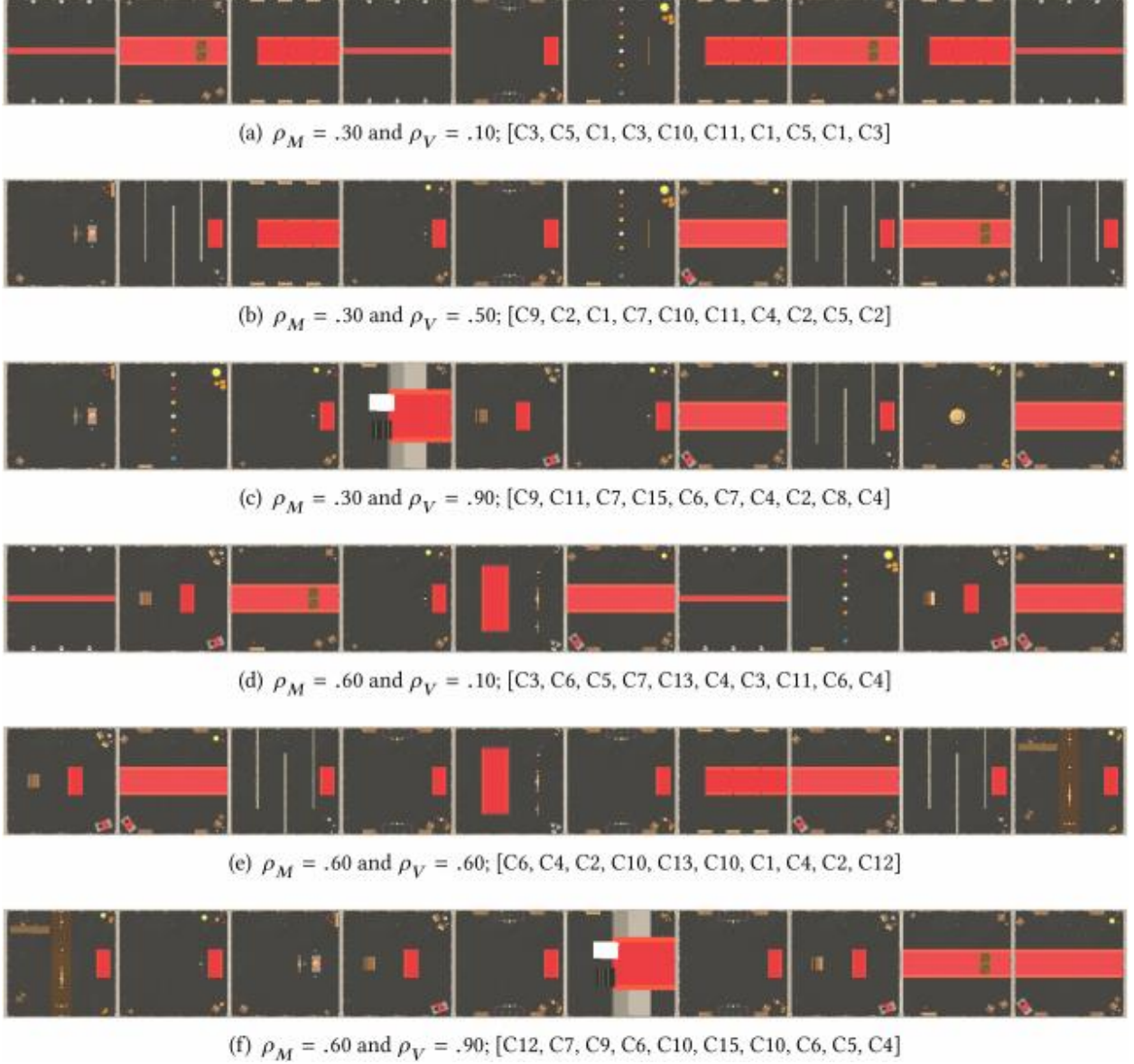


Figure 21: Different game levels synthesized by our system by varying the targets of our cost terms. For all examples, we set the weights of the collaboration-related cost terms at $w_{Collab}^M = 1.00$, $w_{Collab}^V = .30$, and $w_{Collab}^P = .50$, and those of the prior cost terms at $w_{Prior}^S = 1.00$ and $w_{Prior}^R = .50$. The same game level chunk can appear more than once at a synthesized level (e.g., C1, C3, and C5 in Figure 21(a)); however, due to the adjacent repetition cost term, the system does not repeat the same chunk one after the other.

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:

$$C_{Collab}^V(L) = \left| \frac{1}{|L|} \sum_{c_i} (\mathcal{D}(c_i) - \bar{\mathcal{D}})^2 - \rho_V \right| \quad (19)$$

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:

$$C_{Collab}^P(L) = \frac{1}{|L|} \sum_{c_i} \left(\mathcal{N}(\mathcal{D}(c_i)) - \mathcal{N}(\mathcal{D}(g_i)) \right)^2 \quad (20)$$

where g_i is the target degree of collaboration for the $i - th$ game level chunk from the predefined 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 Figure 22 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 species 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.

3.4.3 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:

$$C_{Prior}^S(L) = 1 - \exp\left(-\frac{1}{2\sigma_S^2}(|L| - \rho_S)^2\right) \quad (21)$$

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:

$$C_{Prior}^R(L) = \frac{1}{|L|-1} \sum_{c_i, c_{i+1}} \Gamma(c_i, c_{i+1}) \quad (22)$$

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.

3.4.4 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:

$$f(L) = \exp\left(-\frac{1}{t}C_{Total}(L)\right) \quad (23)$$

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 $p_{add} = .40$, “remove a game level chunk” as $p_{remove} = .20$, and “replace a game level chunk” as $p_{replace} = .40$. 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, $C_{Total}(L')$, with the total cost value, $C_{Total}(L)$, 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:

$$p_{add}(L'|L) = \min\left(1, \frac{p_{remove}}{p_{add}} \frac{U - |L|}{|L'|} \frac{f(L')}{f(L)}\right) \quad (24)$$

the probability for the “remove a game level chunk” move as:

$$p_{remove}(L'|L) = \min\left(1, \frac{p_{add}}{p_{remove}} \frac{|L|}{U - |L'|} \frac{f(L')}{f(L)}\right) \quad (25)$$

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

$$p_{replace}(L'|L) = \min\left(1, \frac{f(L')}{f(L)}\right) \quad (26)$$

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 $C_{Total}(L)$ 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 $w_{Collab}^M = 1.00$, $w_{Collab}^V = .30$, and $w_{Collab}^P = .50$. For the prior cost terms, we set the weights at $w_{Prior}^S = 1.00$ and $w_{Prior}^R = .50$. We assign a high weight value to w_{Collab}^M 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_{Prior}^S 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_{Prior}^S , 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_{Collab}^V , w_{Collab}^P , and w_{Prior}^R 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. Figure 21 illustrates the examples of the synthesized game levels with different targets for the collaboration cost terms. Figure 22 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.

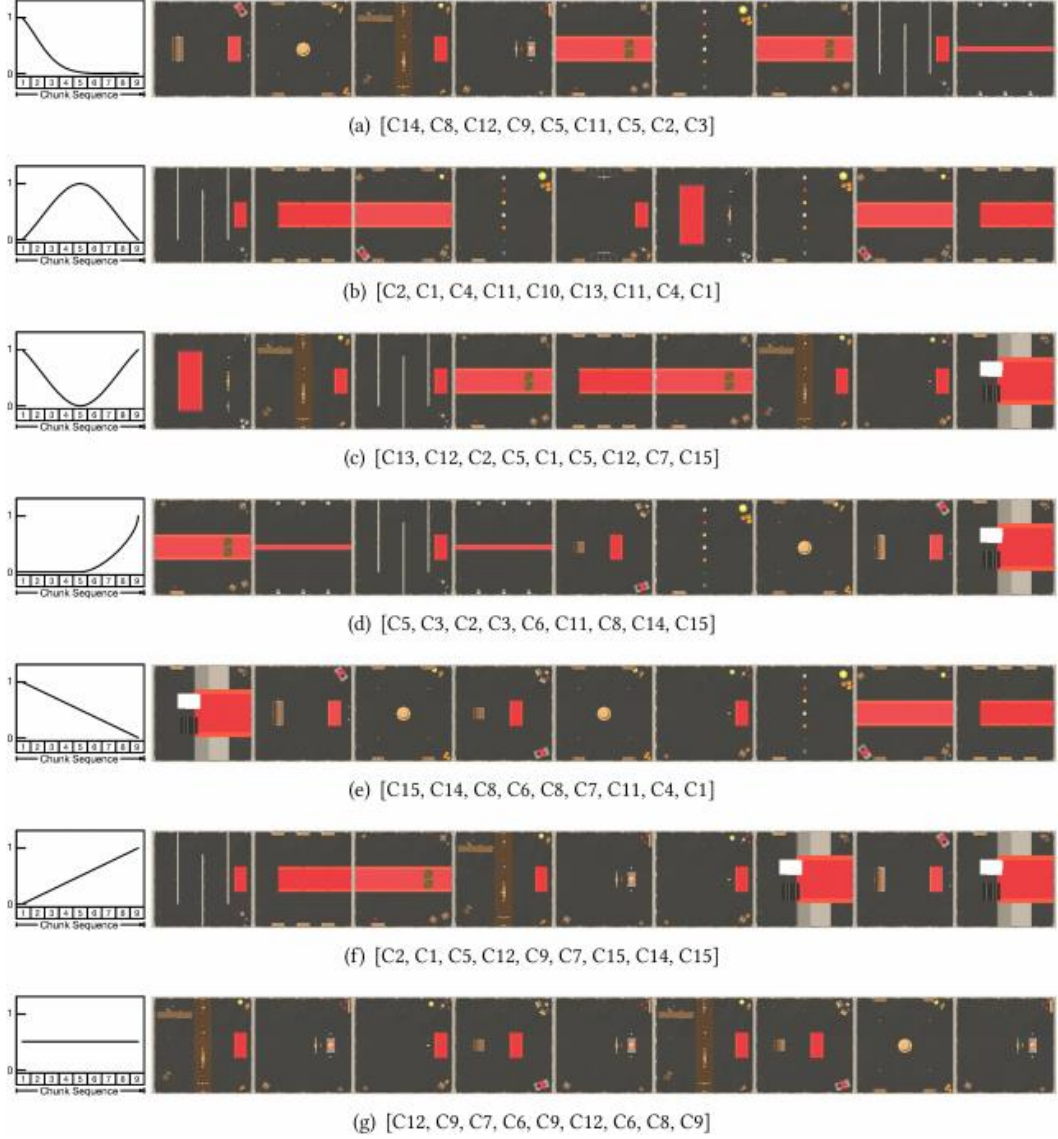


Figure 22: Example game levels ($\rho_S = 9$) using different degrees of collaboration progress line graphs while maintaining the mean degree of collaboration target constant. For all examples, we use $\rho_M = .50$ and $\rho_V = .50$ as the targets.

3.5 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.

3.5.1 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, medium-to-large effect size of $f = .35$ [22], non-sphericity correction $\epsilon = .70$, 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.

3.5.2 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 Networking²⁸ asset to enable the networking functionality between the two computers and, consequently, to allow the participants to collaborate in a shared virtual space.

²⁸ <https://www.photonengine.com/pun>.

3.5.3 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).

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 Figure 22(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 and 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. Figure 23 shows the three synthesized game levels we used in our study. The LC game level (Figure 23(a)) indicated that such a game level is mainly composed of low collaboration activity game level chunks, the MC game level (Figure 23(b)) is primarily formed by medium collaboration activity game level chunks, and the HC game level (Figure 23(c)) is mainly composed of medium and high collaboration activity game level chunks.

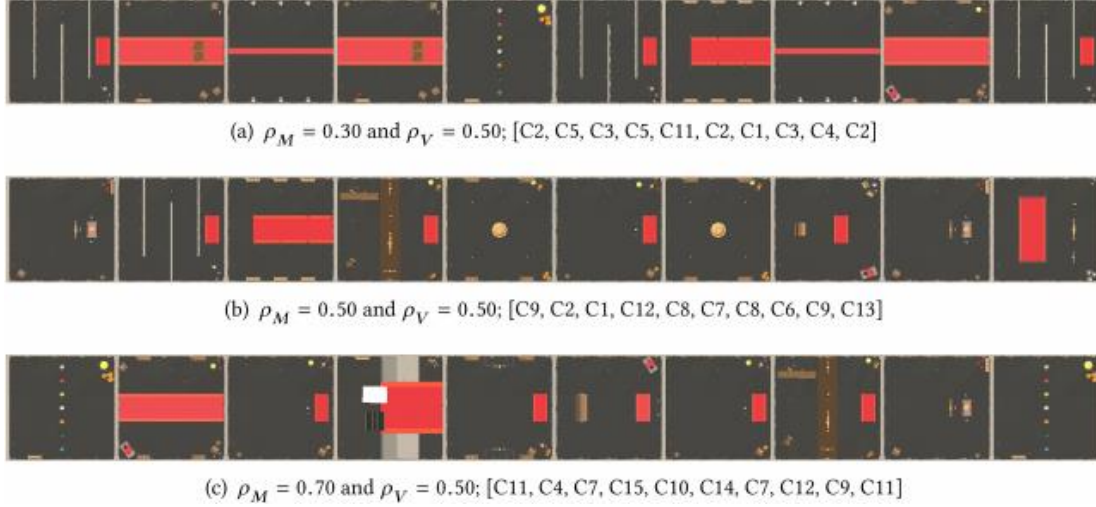


Figure 23: Three different synthesized game levels used in our study. From top to bottom: (a) low degree of collaboration, (b) medium degree of collaboration, and (c) high degree of collaboration.

3.5.4 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.

3.5.5 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 (Figure 18). 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 Figure 23 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.

3.5.6 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.

3.5.6.1 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 (<i>p</i> = .001)
MC	.40	.03	.32	.47	MC<HC (<i>p</i> = .001)
HC	.45	.04	.38	.55	LC<HC (<i>p</i> = .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 (<i>p</i> = .001)
MC	717.40	58.20	638.04	832.21	MC<HC (<i>p</i> = .007)
HC	799.19	93.41	611.40	969.68	LC<HC (<i>p</i> = .0001)
Completion Time (in sec)					
LC	110.73	16.54	85.34	143.86	LC<MC (<i>p</i> = .001)
MC	146.15	24.61	90.46	191.09	MC<HC (<i>p</i> = .002)
HC	178.91	31.70	112.67	236.13	LC<HC (<i>p</i> = .001)
Collaboration Time (in sec)					
LC	22.72	5.86	11.40	35.64	LC<MC (<i>p</i> = .001)
MC	59.16	9.28	41.24	77.64	MC<HC (<i>p</i> = .001)
HC	84.97	17.81	58.77	140.89	LC<HC (<i>p</i> = .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	

3.5.6.2 Subjective Ratings.

The perceived collaboration was also statistically significant across the examined conditions ($\Lambda = .469$, $F [2, 23] = 27.145$, $p = .0001$, $\eta_p^2 = .231$). 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.

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	

3.5.6.3 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 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).

Chu nk ID	Degree of Collabora- tion		Player Distance		Travel Distance		Completion Time		Collaboration Time		Close Proximity Time	
	P	AI	P	AI	P	AI	P	AI	P	AI	P	AI
C1	.021 68	.216 59	11.53 515	3.840 15	41.843 16	37.368 40	5.860 87	10.05 723	.0000 0	1.053 40	.021 68	.035 01
C2	.074 66	.211 31	8.183 53	5.132 70	95.318 88	109.64 185	15.44 542	31.55 432	2.500 24	5.651 69	.072 39	.004 61
C3	.297 78	.217 44	12.42 533	.5889 4	41.680 53	40.081 04	5.686 82	10.00 639	1.566 06	2.253 98	.020 89	.084 87
C4	.242 46	.327 82	9.584 11	5.495 95	46.831 38	43.567 03	7.315 20	11.35 065	1.781 27	4.446 27	.031 68	.076 20
C5	.303 32	.273 82	10.32 784	2.276 46	48.709 96	36.087 67	8.453 67	9.458 78	2.529 87	2.782 06	.034 38	.040 00
C6	.594 34	.515 31	4.603 91	.6281 3	53.818 59	40.156 09	14.56 040	10.04 309	8.432 10	5.458 90	.029 10	.069 98
C7	.524 61	.495 80	11.64 476	1.946 19	54.154 56	43.553 83	10.66 392	11.60 944	5.559 07	6.678 65	.041 41	.066 98
C8	.691 14	.520 15	12.45 848	12.33 019	89.226 49	81.918 81	16.36 952	26.34 777	11.36 309	12.64 062	.031 46	.004 31
C9	.637 97	.459 49	9.629 62	3.211 42	68.365 94	45.389 21	14.48 255	15.63 601	9.256 26	7.161 66	.048 72	.019 37
C10	.653 18	.704 75	17.79 996	14.54 406	208.75 490	106.59 528	44.00 969	77.62 875	28.12 076	46.55 848	.051 04	.043 15
C11	.123 35	.403 82	11.88 042	14.28 864	82.990 25	93.581 84	21.11 941	29.03 274	2.562 56	9.438 18	.055 24	.000 00
C12	.097 61	.433 50	8.618 64	10.38 947	65.356 83	68.061 72	13.91 274	25.77 981	1.153 55	8.102 78	.030 64	.017 29
C13	.139 13	.773 91	16.44 345	12.89 808	93.894 79	67.263 03	26.08 352	25.73 684	3.654 88	13.28 221	.029 49	.013 05
C14	.780 93	.714 62	11.23 880	12.58 725	66.392 09	41.048 68	19.88 709	10.83 099	15.90 273	5.639 66	.021 14	.004 74
C15	.783 48	.769 37	12.34 321	4.752 34	68.783 22	66.579 27	15.00 534	17.69 756	11.71 757	16.40 833	.012 60	.003 99

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 ($r = -.033, n = 15, p = .902$).

3.5.7 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 $w_{Collab}^M = 1.00$, 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 ($\rho_M = .30$ for the LC, $\rho_M = .50$ for the MC, and $\rho_M = .70$ 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, 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.

3.6 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 Figure 22, that some chunks (e.g., C15 in Figure 22(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 ²⁹) and soccer games (such as FIFA²⁹) 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 Explodes¹²). 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

²⁹ [https://en.wikipedia.org/wiki/FIFA_\(video_game_series\)](https://en.wikipedia.org/wiki/FIFA_(video_game_series)).

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.

3.7 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.

3.8 References

- [1] Adler, Paul. S. and Kwon, Seok-Woo. 2002. Social capital: Prospects for a new concept. *Academy of management review*, 27(1), 17-40.
- [2] Alharthi, Sultan. A., Raptis, George. E., Katsini, Christina., Dolgov, Igor., Nacke, Lennart. E. and Toups, Z. O. 2021. Investigating the effects of individual cognitive styles on collaborative gameplay. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 28(4), 1-49.
- [3] Alharthi, Sultan. A., Torres, Ruth. C., Khalaf, Ahmed. S., Toups, Zachary. O., Dolgov, Igor. and Nacke, Lennart. E. 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* (pp. 1-13).
- [4] Amato, Alba. 2017. Procedural content generation in the game industry. In *Game Dynamics: Best Practices in Procedural and Dynamic Game Content Generation*. Springer, New York City, New York, USA, 15-25.
- [5] Bauer, Aaron. and Popović, Zoran. 2012. RRT-based game level analysis, visualization, and visual refinement. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (Vol. 8, No. 1, pp. 8-13).
- [6] Berseth, Glen., Haworth, M. Brandon., Kapadia, Mubbasir. and Faloutsos, Petros. 2014. Characterizing and optimizing game level difficulty. In *Proceedings of the 7th International Conference on Motion in Games* (pp. 153-160).
- [7] Berseth, Glen., Haworth, M. Brandon., Kapadia, Mubbasir. and Faloutsos, Petros. 2014. Characterizing and optimizing game level difficulty. In *Proceedings of the 7th International Conference on Motion in Games* (pp. 153-160).
- [8] Berseth, Glen., Kapadia, Mubbasir. and Faloutsos, Petros. 2013. Steerplex: Estimating scenario complexity for simulated crowds. In *Proceedings of Motion in Games* (pp. 67-76).
- [9] Biocca, Frank., Harms, Chad. and Gregg, Jenn. 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 (pp. 1-9).

- [10] Buchinger, Diego. and da Silva Hounsell, Marcelo. 2018. Guidelines for designing and using collaborative-competitive serious games. *Computers & Education*, 118, 133-149.
- [11] Cavallo, Justin. V., Holmes, John. G., Fitzsimons, Grainne. M., Murray, Sandra. L. and Wood, Joanne. V. 2012. Managing motivational conflict: How self-esteem and executive resources influence self-regulatory responses to risk. *Journal of personality and social psychology*, 103(3), 430.
- [12] Cheung, Victor., Chang, Y. L. Betty. and Scott, Stacey. D. 2012. Communication channels and awareness cues in collocated collaborative time-critical gaming. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work* (pp. 569-578).
- [13] Chib, Siddhartha. and Greenberg, Edward. 1995. Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4), 327-335.
- [14] Chou, Huey-Wen. 2001. Influences of cognitive style and training method on training effectiveness. *Computers & Education*, 37(1), 11-25.
- [15] Cohen, Jacob. 2013. *Statistical power analysis for the behavioral sciences*. Academic press. Cambridge, MA.
- [16] Colledanchise, Michele., & Ögren, Petter. 2018. Behavior trees in robotics and AI: An introduction. CRC Press. Boca Raton, Florida.
- [17] Corrigan, Siobhan., Zon, G. D. R., Maij, A., McDonald, Nick. and Mårtensson, L. 2015. An approach to collaborative learning and the serious game development. *Cognition, Technology & Work*, 17, 269-278.
- [18] Danby, Susan., Evaldsson, Ann-Carita., Melander, Helen. and Aarsand, Pal. 2018. Situated collaboration and problem solving in young children's digital gameplay. *British Journal of Educational Technology*, 49(5), 959-972.
- [19] Silva Fernando de Mesentier, Lee, Scott., Togelius, Julian. and Nealen, Andy. 2017. AI-based playtesting of contemporary board games. In *Proceedings of the 12th International Conference on the Foundations of Digital Games* (pp. 1-10).
- [20] Donalek, Ciro., Djorgovski, S. George., Cioc, Alex., Wang, Anwell., Zhang, Jerry., Lawler, Elizabeth., Stacy, Yeh., Ashish, Mahabal., Matthew, Graham., Andrew, Drake., Davidoff, Scott., Norris, Jeffrey. S. and Longo, Giuseppe. 2014. Immersive and collaborative data visualization using virtual reality platforms. In *2014 IEEE International Conference on Big Data (Big Data)* (pp. 609-614). IEEE.
- [21] Fan, Yanan. and Sisson, Scott. A. 2011. Reversible jump MCMC. *Handbook of Markov Chain Monte Carlo*, 67-92.

- [22] Faul, Franz., Erdfelder, Edgar., Buchner, Axel. and Lang, Albert-Georg. 2009. Statistical power analyses using G* Power 3.1: Tests for correlation and regression analyses. *Behavior research methods*, 41(4), 1149-1160.
- [23] Faul, Franz., Erdfelder, Edgar., Lang, Albert-Georg. and Buchner, Axel. 2007. G* Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences. *Behavior research methods*, 39(2), 175-191.
- [24] Ford, Nigel. 2000. Cognitive styles and virtual environments. *Journal of the American Society for information science*, 51(6), 543-557.
- [25] Ford, Nigel. and Chen, Sherry. Y. 2001. Matching/mismatching revisited: An empirical study of learning and teaching styles. *British Journal of Educational Technology*, 32(1), 5-22.
- [26] Francis, Anthony. 2019. Overcoming Pitfalls in Behavior Tree Design. In *Game AI Pro 360: Guide to Architecture*. CRC Press, Boca Raton, Florida, 309–320.
- [27] Gisslén, Linus., Eakins, Andy., Gordillo, Camilo., Bergdahl, Joakim. and Tollmar, Konrad. 2021. Adversarial reinforcement learning for procedural content generation. In *IEEE Conference on Games (CoG)*. IEEE, New York City, New York, USA. 1-8.
- [28] González-Duque, Miguel., Palm, Rasmus. B., Ha, David. and Risi, Sebastian. 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] Gordillo, Camilo., Bergdahl, Joakim., Tollmar, Konrad. and Gisslén, Linus. 2021. Improving playtesting coverage via curiosity driven reinforcement learning agents. In *2021 IEEE Conference on Games (CoG)* (pp. 1-8). IEEE.
- [30] Green, Peter. J. 1995. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4), 711-732.
- [31] Greenwald, Scott. W., Corning, Wiley. and Maes, Pattie. 2017. Multi-user framework for collaboration and co-creation in virtual reality. *12th International Conference on Computer Supported Collaborative Learning (CSCL)*. ACM, 1-2.
- [32] Güttler, Christian. and Johansson, Troels. D. 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] Hendrikx, Mark., Meijer, Sebastiaan., Van Der Velden, Joeri. and Iosup, Alexandru. 2013. Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 9(1), 1-22.

- [34] Ibayashi, Hikaru., Sugiura, Yuta., Sakamoto, Daisuke., Miyata, Natsuki., Tada, Mitsunori., Okuma, Takashi., Takeshi, Kurata., Masaaki, Mochimaru. and Igarashi, Takeo. 2015. Dollhouse VR: a multi-view, multi-user collaborative design workspace with VR technology. In *SIGGRAPH Asia 2015 Emerging Technologies*. ACM, New York, NY, USA, 1-2.
- [35] Kao, Dominic., Magana, Alejandra. J. and Mousas, Christos. 2021. Evaluating tutorial-based instructions for controllers in virtual reality games. *Proceedings of the ACM on Human-Computer Interaction*, 5(*CHI PLAY*), 1-28.
- [36] Keedwell, A. Donald. and Dénes, Jozsef. 2015. *Latin squares and their applications*. Elsevier. Amsterdam, Netherlands.
- [37] Li, Wanwan., Xie, Biao., Zhang, Yongqi., Meiss, Walter., Huang, Haikun. and Yu, Lap-Fai. 2020. Exertion-aware path generation. *ACM Trans. Graph.*, 39(4), 1-14.
- [38] Liapis, Antonios., Yannakakis, Georgios. and Togelius, Julian. 2013. Towards a generic method of evaluating game levels. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. (Vol. 9, No. 1, pp. 30-36).
- [39] Liu, Can., Chapuis, Olivier., Beaudouin-Lafon, Michel. and Lecolinet, Eric. 2016. Shared interaction on a wall-sized display in a data manipulation task. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (pp. 2075-2086).
- [40] Liu, Huimin., Wang, Zhiquan., Mazumdar, Angshuman. and Mousas, Christos. 2021. Virtual reality game level layout design for real environment constraints. *Graphics and Visual Computing*, 4, 200020.
- [41] Liu, Huimin., Wang, Zhiquan., Mousas, Christos. and Kao, Dominic. 2020. Virtual reality racket sports: Virtual drills for exercise and training. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (pp. 566-576). IEEE.
- [42] Ma, Chongyang., Vining, Nicholas., Lefebvre, Sylvain. and Sheffer, Alla. 2014. Game level layout from design specification. In *Computer Graphics Forum* (Vol. 33, No. 2, pp. 95-104).
- [43] Malik, Ali. A., Masood, Tariq. and Bilberg, Arne. 2020. Virtual reality in manufacturing: immersive and collaborative artificial-reality in design of human-robot workspace. *International Journal of Computer Integrated Manufacturing*, 33(1), 22-37.
- [44] Melis, Alicia. P., Hare, Brian. and Tomasello, Michael. 2006. Chimpanzees recruit the best collaborators. *Science*, 311(5765), 1297-1300.
- [45] Men, Liang. and Bryan-Kinns, Nick. 2018. LeMo: supporting collaborative music making in virtual reality. In *2018 IEEE 4th VR workshop on sonic interactions for virtual environments (SIVE)*. IEEE, New York, NY, USA, 1-6.

- [46] Merrick, Kathryn. E., Isaacs, Amitay., Barlow, Michael. and Gu, Ning. 2013. A shape grammar approach to computational creativity and procedural content generation in massively multiplayer online role playing games. *Entertainment Computing*, 4(2), 115-130.
- [47] Mora-Cantalops, Marcal. and Sicilia, Miguel-Ángel. 2018. MOBA games: A literature review. *Entertainment computing*, 26, 128-138.
- [48] Mousas, Christos., Krogmeier, Claudia. and Wang, Zhiquan. 2021. Photo Sequences of Varying Emotion: Optimization with a Valence-Arousal Annotated Dataset. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 11(2), 1-19.
- [49] Piumsomboon, Thammathip., Lee, Youngho., Lee, Gun. and Billinghamurst, Mark. 2017. CoVAR: a collaborative virtual and augmented reality system for remote collaboration. In *SIGGRAPH Asia 2017 Emerging Technologies*. ACM, New York, NY, USA, 1-2.
- [50] Reinders, Hayo. and Wattana, Sorada. 2014. Can I say something? The effects of digital game play on willingness to communicate. *Language Learning & Technology*. 18, 2, 101–123.
- [51] Reuter, Christian. 2016. “Authoring Collaborative Multiplayer Games-Game Design Patterns, Structural Verification, Collaborative Balancing and Rapid Prototyping”. Ph.D. Dissertation. Technische Universität Darmstadt.
- [52] Reuter, Christian., Wendel, Viktor., Göbel, Stefan. and Steinmetz, Ralf. 2014. Game Design Patterns for Collaborative Player Interactions. In *DiGRA*. DiGRA, 1–16.
- [53] Rocha, Jose. B., Mascarenhas, Samuel. and Prada, Rui. 2008. Game mechanics for cooperative games. *ZON Digital Games*. ZON, 72-80.
- [54] Roohi, Shaghayegh., Guckelsberger, Christian., Relas, Asko., Heiskanen, Henri., Takatalo, Jari. and Hämäläinen, Perttu. 2021. Predicting game difficulty and engagement using AI players. *Proceedings of the ACM on Human-Computer Interaction*, 5(CHI PLAY), 1-17.
- [55] Roohi, Shaghayegh., Relas, Asko., Takatalo, Jari., Heiskanen, Henri. and Hämäläinen, Perttu. 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] Savitsky, Kenneth., Van Boven, Leaf., Epley, Nicholas. and Wight, Wayne. M. 2005. The unpacking effect in allocations of responsibility for group tasks. *Journal of Experimental Social Psychology*, 41(5), 447-457.
- [57] Schell, Jesse. 2008. *The Art of Game Design: A book of lenses*. CRC press, New York, NY, USA
- [58] Sedano, Carolina. I., Carvalho, Maira. B., Secco, Nicola. and Longstreet, C. Shaun. 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] Seif El-Nasr, Magy., Aghabeigi, Bardia., Milam, David., Erfani, Mona., Lameman, Beth., Maygoli, Hamid. and Mah, Sang. 2010. Understanding and evaluating cooperative games. In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, New York, NY, USA, 253-262.
- [60] Sekhavat, Yoonas. A. 2017. Behavior trees for computer games. *International Journal on Artificial Intelligence Tools*, 26(02), 1730001.
- [61] Shoulson, Alexander., Gilbert, Max. L., Kapadia, Mubbasir. and Badler, Norman. I. 2013. An event-centric planning approach for dynamic real-time narrative. In *Proceedings of Motion on Games*. ACM, New York, NY, USA, 121-130.
- [62] Siu, Ka-Chun., Best, Bradley. J., Kim, Jong. Wook., Oleynikov, Dmitry. and Ritter, Frank. E. 2016. Adaptive virtual reality training to optimize military medical skills acquisition and retention. *Military medicine*, 181(suppl_5), 214-220.
- [63] Slater, Mel., Usoh, Martin. and Steed, Anthony. 1994. Depth of presence in virtual environments. *Presence: Teleoperators & Virtual Environments*, 3(2), 130-144.
- [64] Smith, Adam. M. and Mateas, Michael. 2011. Answer set programming for procedural content generation: A design space approach. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3), 187-200.
- [65] Sugden, Robert. 2015. Team reasoning and intentional cooperation for mutual benefit. *Journal of Social Ontology*, 1(1), 143-166.
- [66] Tang, Anthony., Tory, Melanie., Po, Barry., Neumann, Petra. and Carpendale, Sheelagh. 2006. Collaborative coupling over tabletop displays. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, New York, NY, USA, 1181-1190.
- [67] Thomson, Ann. Marie., Perry, James. L. and Miller, Theodore. K. 2009. Conceptualizing and measuring collaboration. *Journal of public administration research and theory*, 19(1), 23-56.
- [68] Shaker, Noor., Togelius, Julian. and Nelson, Mark. J. 2016. *Procedural content generation in games: A Textbook and an Overview of Current Research*. Springer, Berlin, Germany.
- [69] Togelius, Julian., Yannakakis, Georgios. N., Stanley, Kenneth. O. and Browne, Cameron. 2011. Search-based procedural content generation: A taxonomy and survey. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3), 172-186.
- [70] Tsai, Wenpin. and Ghoshal, Sumantra. 1998. Social capital and value creation: The role of intrafirm networks. *Academy of management Journal*, 41(4), 464-476.
- [71] Tyack, April., Wyeth, Peta. and Johnson, Daniel. 2016. The appeal of MOBA games: What makes people start, stay, and stop. In *Proceedings of the 2016 annual symposium on computer-human interaction in play*. ACM, New York, NY, USA, 313-325.

- [72] Uhlaner, Lorraine. M., Matser, Ilse. A., Berent-Braun, Marta. M. and Flören, Roberto. H. 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), 260-277.
- [73] van Arkel, Benjamin., Karavolos, Daniel., Bouwer, Anders., Bakkes, Sander. and Nack, Frank. 2015. Procedural generation of collaborative puzzle-platform game levels. In *International Conference on Intelligent Games and Simulation*. Eurosis, 87-93.
- [74] van Rozen, Riemer. and Heijn, Quinten. 2018. Measuring quality of grammars for procedural level generation. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*. ACM, New York, NY, USA, 1-8.
- [75] Wang, Tong. and Kurabayashi, Shuichi. 2020. Sketch2map: a game map design support system allowing quick hand sketch prototyping. In *2020 IEEE Conference on Games (CoG)*. IEEE, New York, NY, USA, 596-599.
- [76] Wheat, Daniel., Masek, Martin., Lam, Chiou. Peng. and Hingston, Philip. 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] Winn, Brain. M. and Fisher, J. W. 2004. Design of communication, competition, and collaboration in online games. In *Dipresentasikan Dalam Computer Game Technology Conference*. Citeseer, 1-13.
- [78] Wuertz, Jason., Alharthi, Sultan. A., Hamilton, William. A., Bateman, Scott., Gutwin, Carl., Tang, Anthony., Zachary, Touns. and Hammer, Jessica. 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] Xie, Biao., Liu, Huimin., Alghofaili, Rawan., Zhang, Yongqi., Jiang, Yeling., Lobo, Flavio. Destri., Li, Changyang., Li Wanwan., Huang, Haikun., Akdere, Mesut., Mousas, Christos. and Yu, Lap-Fai. 2021. A review on virtual reality skill training applications. *Frontiers in Virtual Reality*, 2, 645153.
- [80] Xie, Biao., Zhang, Yongqi., Huang, Haikun., Ogawa, Elisa., You, Tongjian. and Yu, Lap-Fai. 2018. Exercise intensity-driven level design. *IEEE transactions on visualization and computer graphics*, 24(4), 1661-1670.
- [81] Zagal, Jose. P., Rick, Jochen. and Hsi, Idris. 2006. Collaborative games: Lessons learned from board games. *Simulation & gaming*, 37(1), 24-40.
- [82] Zea, Natalia. Padilla., Sánchez, Jose. Luis. Gonzalez., Gutiérrez, Francisco. L., Cabrera, Marcelino. J. and Paderewski, Patricia. 2009. Design of educational multiplayer videogames: A vision from collaborative learning. *Advances in Engineering Software*, 40(12), 1251-1260.

- [83] Zhang, Li-Fang. 2006. Does student–teacher thinking style match/mismatch matter in students’ achievement?. *Educational psychology*, 26(3), 395-409.
- [84] Zhou, Zhuoming., Márquez Segura, Elena., Duval, Jared., John, Michael. and Isbister, Katherine. 2019. Astaire: A collaborative mixed reality dance game for collocated players. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. ACM, New York, NY, USA, 5-18.
- [85] Zohaib, Mohammad. 2018. Dynamic Difficulty Adjustment (DDA) in computer games: A review. *Advances in Human-Computer Interaction*, 2018.

CHAPTER 4. PUBLISHED ARTICLE #4: SYNTHESIZING SHARED SPACE VIRTUAL REALITY FIRE EVACUATION TRAINING DRILLS

All the authors in the paper agreed on the author of the dissertation to use this publication in her Ph.D. dissertation.

This paper was published previously in *International Symposium on Mixed and Augmented Reality (Adjunct) (ISMAR Adjunct) 2022*.

© [2022] IEEE. Reprinted, with permission, from [Liu, Huimin., Choi, Minsoo., Yu, Liuchuan., Koiliias, Alexandros., Yu, Lai-Fai. and Mousas, Christos, Synthesizing Shared Space Virtual Reality Fire Evacuation Training Drills. 2022 *IEEE International Symposium on Mixed and Augmented Reality (Adjunct) (ISMAR Adjunct)*, 2022, October]

(Cite) Liu, H., Choi, M., Yu, L., Koiliias, A., Yu, L. F., & Mousas, C. Synthesizing Shared Space Virtual Reality Fire Evacuation Training Drills. In 2022 *IEEE International Symposium on Mixed and Augmented Reality (Adjunct) (ISMAR Adjunct)* (pp. 459-464). IEEE. DOI: [10.1109/ISMAR-Adjunct57072.2022.00097](https://doi.org/10.1109/ISMAR-Adjunct57072.2022.00097)

The authors confirm contribution to the paper as follows:

- Study Conceptualization and Design: Mousas, Christos and Yu, Lai-Fai;
- Methodology and/or Simulation Design: Mousas, Christos; Choi, Minsoo and Liu, Huimin;
- Execution/Implementation: Liu, Huimin; Choi, Minsoo; Yu, Liuchuan and Koiliias, Alexandros;
- Data collection: Liu, Huimin and Yu, Liuchuan;
- Analysis and interpretation of results: Mousas, Christo and Liu, Huimin;
- Findings, Conclusions, and/or Recommendations: Mousas, Christos and Liu, Huimin;
- Draft manuscript preparation: Mousas, Christos; Liu, Huimin; Choi, Minsoo; Yu, Liuchuan; and Yu, Lai-Fai.

All authors reviewed the results and approved the final version of the manuscript.

All authors in this paper agree on the usage of the paper in this dissertation.

Purdue does not require any formal permission.

Abstract

We synthesized virtual reality fire evacuation training drills in a shared virtual space to explore people's collaboration behavior. We formulate the authoring process of the fire evacuation training drill in a total cost function, which we later solve with a Markov Chain Monte Carlo (MCMC) optimization-based method. The users' assigned task in the synthesized training drill is to help virtual agents evacuate the building as quickly as possible using predefined interaction mechanisms. The users can join the training drill from different physical locations and collaborate and communicate in a shared virtual space to finish the task. We conducted a user study to collect both in-game measurements and subjective ratings to evaluate whether the synthesized training drills would affect how the participants collaborated.

4.1 Introduction

Collaboration is usually characterized by shared goals, group activities, communication, and exchanging information [17]. Roschelle and Teasley [26] defined collaboration on a joint problem space as the "mutual engagement of people in a coordinated effort to solve a problem together." Various researchers [5][8] regard collaboration as an essential component of effective training and learning in comparison to individual tasks. In the age of fast-paced development of globalization, which has a higher requirement for productivity, especially during the COVID-19 pandemic when people have been impeded from meeting in person, the importance of remote collaboration systems has been emphasized, as they contribute to remote team task success, reduce travel expenses, ensure safety, reduce carbon emissions, increase efficiency, and save time and energy.

However, the concept of collaboration is abstract and difficult to grasp [11], making it challenging to utilize in practical applications. When implementing collaborative training scenarios in virtual environments, designers usually manually build the contents according to their subjective experiences and intuition in order to trigger the intended behavior in participants. This process is tedious and time-consuming since it lacks a solid theory that supports the effectiveness of the designed content. To better support collaboration on common tasks among the involved group members, it is necessary to obtain a more precise understanding of collaboration and how

to conduct immersive collaboration remotely in a shared virtual space using modern virtual reality (VR) technologies.

The project presented in this paper focused on synthesizing VR fire evacuation training drills in a shared virtual space to explore the participants' collaboration behavior. Inspired by procedural content generation approaches, we proposed an optimization-based method that automatically generates fire evacuation training drills with varying levels of difficulty. The users' assigned task is to help virtual agents evacuate the building as quickly as possible using predefined interaction mechanisms (voice commands, trigger fire extinguisher, physical locomotion, etc.). The participants can join the training drill from different locations and collaborate and communicate in a shared virtual space to accomplish the task (see Figure 24). We evaluated the proposed VR training drill authoring method by conducting a user study among three training drills with different difficulty levels: low difficulty (LD), medium difficulty (MD), and high difficulty (HD). We collected both in-game measurements and subjective ratings to explore how the participants collaborate in such a VR setup.



Figure 24: Two players in different locations, wearing a VR headset on the VR treadmill. Their task is to guide the agents out of the building where a simulated fire emergency occurs. The two players are in the same virtual space even though their physical locations are different. They can communicate, use voice commands to guide the agents outside the building, and use a fire extinguisher to eliminate the fire in the building. We illustrate users' and agents' positions and the top view of the building in the minimap. © [2022] IEEE

4.2 Related Work

Virtual reality (VR) and augmented reality (AR) are as effective a training mechanism as the commonly accepted methods [15]. VR can enhance learning and training. Some work focused on training for sports [20] and education [10]. Also, some research was conducted for medical and rehabilitation purposes [27], and for evacuation training and research purposes [19]. As for AR training, research shows that AR, applied in education and training, has positive potential for the future of education [18]. Moreover, AR shows great potential and can be applied in many other fields, such as, medical education [3], corporate training [22], healthcare simulation [30], maintenance skills [32], and vocational training [6]. For more details about VR training, please refer to Xie et al. [34].

With network, VR and AR can be applied in remote training and collaboration scenarios. Greenwald et al. [13] explored the immense potential for collaborative VR applications for learning. Some researchers proposed frameworks to support collaboration in virtual environments. For example, MedicalVR [21] is a virtual reality framework and assistive tool for medical environment. It outlines real-time collaboration and human-centered design aspects in modern tele-medicine. Kurillo et al. [16] presented a framework for immersive virtual environment intended for remote collaboration and training of physical activities. For example, Tea et al. [29] developed a multi-user immersive virtual reality application for real-time remote collaboration to enhance design review process. Snow Dome [24], which is a mixed reality remote collaboration application, was developed to support multi-scale interaction for a virtual reality user. Elvezio et al. [9] demonstrated an approach to support remote collaboration in AR and VR by virtual replicas, which allows the remote user to create and manipulate virtual replicas of physical objects in the local environment. Besides framework, system, and application, some research focused on adaptive avatar, Mini-Me [25], and toolkit, ColabAR [31], to promote remote collaboration.

In this paper, we propose an optimization-based method to automatically synthesize shared space VR fire evacuation training drills with different difficulty levels. We also demonstrated how to employ synthesized training drills on a networked VR platform with treadmills to enable remote, collaborative training.

4.3 Preliminary Remarks

4.3.1 System Overview

Figure 24 shows our project’s system overview. Two users are in different physical locations and join the developed training drill, which takes place in a virtual space shared through the Internet. Inside the shared virtual space we have synthesized fire evacuation training drill that are generated by using our optimization-based method. Participants are able to extinguish the fires by using an integrated fire extinguisher that will show up on their hands when they enable it. The users can communicate with each other inside the virtual environment freely through Voice over Internet Protocol (VoIP). We placed virtual agents who can respond to participants’ voice commands and need to be rescued. The participants’ common task is to guide all the agents outside the building.

4.3.2 Environment Representation

We represent the input training environment as an $M \times N$ in size 2D grid ($[c_{1,1}, \dots, c_{M,N}]$ denotes the cells of the generated grid; the resolution of the grid is defined by the designer/trainer). Then, we represent each grid cell ($c_{x,y}$) of the grid as either obstacle (T_{obs}), fire (T_{fire}), or empty (T_{empty}) grid cell.

4.3.3 Virtual Training Environment

We designed a virtual school layout according to specific design and safety regulations³⁰ and standards in the US [2]. We have created several types of classrooms (standard classroom, library, basketball court, theater, restrooms, lockers, etc.) to convey a complete impression of a school. The average size of a classroom is 12×12 m with a height of 3.75 m to ensure that participants can move around fast and freely while avoiding virtual objects/obstacles (desk, chairs, etc.). Finally, we have decided to add a significant number of exits (six in total) to ensure that users can find accessible exits under different conditions and effects that block some or most of them. Figure 25 shows screenshots of the designed virtual environment.

³⁰ <https://www.aps.edu/facilities-design-and-construction/design-standards-and-guidelines>

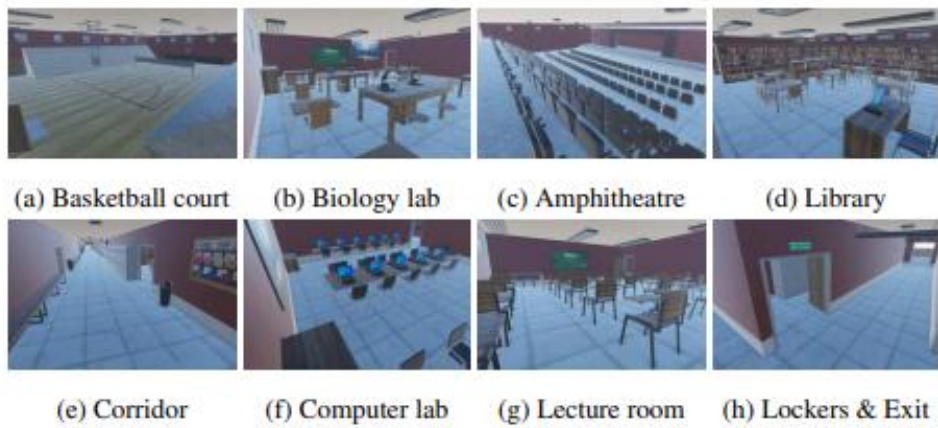


Figure 25: Different parts of the designed virtual environment we used in our prototype application. © [2022] IEEE

The virtual agents can respond to specific voice commands under certain conditions (see Figure 26). There are six usable commands implemented in the system. Among them, we implemented four commands to instruct the agents to move, including “come here,” “follow me,” “run,” and “crawl.” We also included the “stop” and “wait” commands to pause the movement of agents at any time.

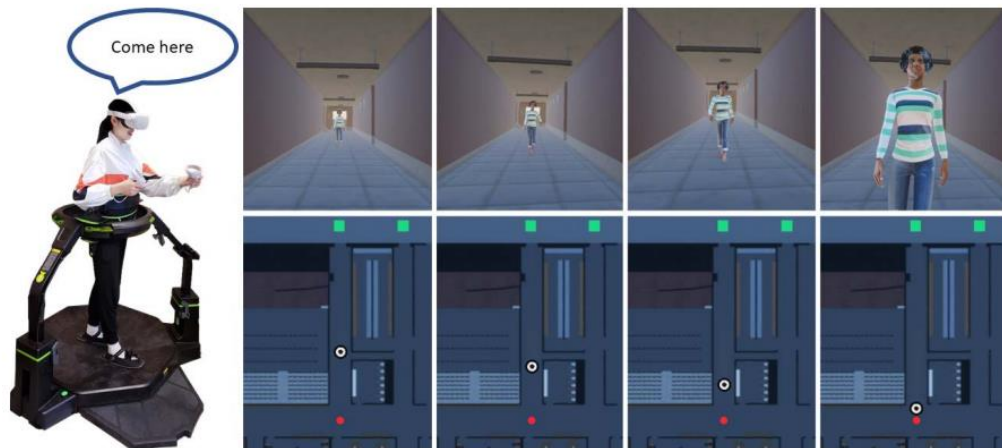


Figure 26: A user commands a virtual agent to “come here” and the agent moves toward the user. © [2022] IEEE

4.3.4 Authoring Training Drill

We represent our training drill as a composition of several fires $F = [f_1, \dots, f_K]$ taking places in a static 3D environment. Their position and size are determined based on our proposed optimization-based method (see Section 4). There are also several trainer-defined virtual agents $A = [a_1, \dots, a_B]$ placed in different locations in the virtual environment. The trainer instructs the users of our training drill to rescue the virtual agents by helping them exit the building.

4.4 Problem Formulation

The design of the evacuation drill d is evaluated by the total cost function $C_{Total}(d)$:

$$C_{Total}(d) = w_{length}C_{length}(d) + w_{turns}C_{turns}(d) + w_{fire}C_{fire}(d) + w_{vis}C_{vis}(d) \quad (27)$$

where C_{length} encodes the length of the optimal path that the user should follow to fulfil the necessary goals and exit the building; C_{turns} encodes the number of turns in the optimal path; C_{fire} denotes the number of fires that the user should extinguish to fulfil the necessary goals (e.g., access the virtual agents, help virtual agents exit the building); and C_{vis} denotes the visibility conditions of the virtual environment. w_{length} , w_{turns} , w_{fire} , and w_{vis} are the corresponding weights of each cost term, prioritized by importance. We discuss the details for each cost term as follows.

Length Cost. The path synthesized by our system represents how far the user must walk in the training environment to execute the required task. The length cost is used to compare the length of the synthesized path against the user-defined target path length. We present this cost as:

$$C_{length}(d) = \frac{1}{L_{diag}} |\sum_{G_i(A)} L(P_i) - \rho_{length}| \quad (28)$$

where L_{diag} is used as a normalization term representing the diagonal length of the entire virtual environment; ρ_{length} denotes the user-defined path length; P_i is the path between each $G_i(A)$ subgroup of agents that are in a specific location (e.g., in the basketball court) in the virtual environment requiring rescue, where $G_i(A) \leq A$; and $L(P_i)$ is the distance between the i -th sub-group of agents $G_i(A)$ and the closest exit in the training environment. To compute the length of the chosen optimal path, we use an improved version of the A* algorithm [35]. For each returned pair of adjacent cells (c_j, c_{j+1}) belonging to the P_i path in the grid, we compute path

length $L(P_i)$ by summing the length of each pair of adjacent cells $\mathcal{L}(c_j, c_{j+1})$ from the optimal path as:

$$L(P_i) = \sum_{c_j, c_{j+1}}^{|P_i|-1} \mathcal{L}(c_j, c_{j+1}) \quad (29)$$

where $|P_i|$ denotes the total number of grid cells from the optimal path. Note that the obstacle T_{obs} and fire T_{fire} grid cells are blocked, and the empty grid cell T_{empty} is unblocked. However, during the optimization process, if there is no optimal path, we label the fire grid cell as unblocked, and therefore it can be considered part of the optimal path. Thus, the synthesized path length comes closer to the target path length and makes the training drill more difficult since the user needs to extinguish a fire to access that path properly.

Turn Cost. The turn cost is used to compare the number of turns in the path against a user-defined target number of total turns ρ_{turns} :

$$C_{turns}(d) = \left| \frac{\sum_{|P|} \mathcal{T}(P_i) - \rho_{turns}}{\rho_{turns}} \right| \quad (30)$$

where $\mathcal{T}(P_i)$ returns the number of turns in the optimal path P_i , and $|P|$ denotes the total number of optimal paths the users should follow to accomplish the task. To calculate $\mathcal{T}(P_i)$, we consider all triads of adjacent grid cells. If these three grid cells do not form a straight line, they are regarded as a turn and, therefore, $\mathcal{T}(P_i)$ returns 1; otherwise, it returns 0.

Fire Cost. Users must extinguish fires to reach virtual agents, access parts of the virtual building, or exit the virtual building. The fire cost compares the number of fires that the user should extinguish against the designer-specified target number of fires ρ_{fire} :

$$C_{fire}(d) = \frac{1}{U} \left| \sum_{|F|} \Gamma(f_i) - \rho_{fire} \right| \quad (31)$$

where $\Gamma(f_i)$ returns 1 if f_i is found to be in the optimal path; otherwise, it returns 0. U is used as a normalization factor representing the upper limit of the number of fires. We set $U = 40$ as the upper limit value for all examples presented in this paper.

Visibility Cost. The user's visibility in the virtual environment is computed by considering the ratio between the area occupied by the fires over the total area of the virtual environment. We compare it against a user-defined target value:

$$C_{vis}(d) = \left| \frac{\sum_{|F|} \mathcal{A}(f_i)}{\mathcal{A}(e)} - \rho_{vis} \right| \quad (32)$$

where $\sum_{|F|} \mathcal{A}(f_i)$ represents the total area occupied by the fires; $\mathcal{A}(e)$ represents the total area of the entire virtual environment; and ρ_{vis} is user-defined target visibility. Note that a high value of $\rho_{vis} \in [0, 1]$ denotes low visibility and vice versa.

4.5 Optimization

To assess all possible training outcomes during the optimization process, our system optimizes total cost functions through the reversible-jump Markov chain Monte Carlo (RJMCMC) method [12]. We apply simulated annealing using a Metropolis-Hastings state-search step [7]. We start by defining a Boltzmann-like objective function:

$$f(d) = \exp\left(-\frac{1}{t} C_{Total}(d)\right) \quad (33)$$

where t encodes the temperature parameter of simulated annealing. During the optimization process, the system proposes a new configuration of the training drill d' by altering the current training drill d using one of the following moves:

- **Adding a fire:** Our system places a randomly sized fire in a randomly chosen position in the virtual environment.
- **Removing an existing fire:** Our system randomly chooses a fire from the virtual environment to remove.
- **Modifying an existing fire:** Our system randomly chooses a fire from the virtual environment and modifies its size and position.

We set the probability of adding a fire as $p_{add} = .40$, the probability of removing a fire as $p_{remove} = .20$, and the probability of modifying a fire as $p_{modify} = .40$. Through these probabilities, our system chooses to add and modify a fire more often than choosing to remove a fire. By applying one of these moves, our system proposes a training drill d' and compares the total cost of the proposed training drill $C_{Total}(d')$ with the total cost of the current training drill $C_{Total}(d)$ to determine whether the system accepts the proposed training drill d' or keeps the current training drill d .

To ensure balanced trans-dimensional optimization, we define the probability of each move. Our system computes the probability of adding a fire as:

$$p_{add}(d'|d) = \min\left(1, \frac{p_{remove}}{p_{add}} \frac{U-|d|}{|d'|} \frac{f(d')}{f(d)}\right) \quad (34)$$

computes the probability of removing an existing fire as:

$$p_{remove}(d'|d) = \min\left(1, \frac{p_{add}}{p_{remove}} \frac{|d|}{U-|d'|} \frac{f(d')}{f(d)}\right) \quad (35)$$

and computes the probability of modifying an existing fire as:

$$p_{modify}(d'|d) = \min\left(1, \frac{f(d')}{f(d)}\right) \quad (36)$$

Based on the above formulation, we set an upper limit on the number of fires during optimization using the variable $U = 40$. Thus, our system synthesizes a virtual environment with fires equal to or less than U .

We also applied simulated annealing to explore our solution space effectively. Simulated annealing allows us to use a temperature parameter t to control the acceptance probability of the proposed training drill d' . If the temperature parameter is high, the system will aggressively explore the whole solution space. If the temperature parameter is low, the optimizer will become more selective. We initialize the temperature parameter as $t = 1.00$ at the beginning of optimization. In each iteration, we multiply the temperature parameter by 0.998. The optimization process terminates when the change in $C_{Total}(d)$ is less than 5% of the previous 50 iterations.

Unless specified otherwise, we set the weight of the length cost to $w_{length} = 1.00$, the weight of the turn cost to $w_{turns} = .40$, the weight of the fire cost to $w_{fire} = .60$, and the weight of the visibility cost to $w_{vis} = .40$. Via those weights, our system prioritizes the length of the path and the number of fires the user must extinguish. However, the designer may change the priority by changing the weights.

4.6 User Study

The user study was conducted between two universities (Purdue and GMU) across states in the US. The two universities were not in the same physical spaces. The intent of our project is to evaluate whether our proposed method can synthesize training drills with different targeted difficulty levels, thus triggering any difference in the collaboration behavior among participants. The methodology of the study is described in the following subsections. Figure 27 shows example scenes from the synthesized training drill.



Figure 27: Example scenes from the synthesized training drill. © [2022] IEEE

4.6.1 Participants

We recruited participants in both universities via class announcements and emails. Participants from each university were randomly assigned to a group. Each group was scheduled to attend the study simultaneously at each location. Participants in the same group remotely joined the shared virtual space to experience the synthesized training drills. We collected data from 27 groups (54 volunteers; 34 male and 20 female). The age of the participants were between 17-30 years ($M = 19.96, SD = 2.88$). All participants have experienced virtual reality before.

4.6.2 Conditions

We developed three experimental conditions to determine whether the optimized training drills with differently targeted difficulty would influence the collaboration behaviors among the participants. The experiment followed a within-group study design. We used the Latin squares [33] ordering method to balance the conditions and minimize the carryover effects. Figure 28 shows the three synthesized training drills used in our experiment. The conditions were as follows:

- Low Difficulty (LD): We set the target cost terms as: $\rho_{length} = 280$, $\rho_{turns} = 30$, $\rho_{fire} = 3$, and $\rho_{vis} = .20$.
- Medium Difficulty (MD): We set the cost terms as: $\rho_{length} = 300$, $\rho_{turns} = 35$, $\rho_{fire} = 5$, and $\rho_{vis} = .50$.
- High Difficulty (HD): We set the cost terms as: $\rho_{length} = 320$, $\rho_{turns} = 40$, $\rho_{fire} = 7$, and $\rho_{vis} = .80$.

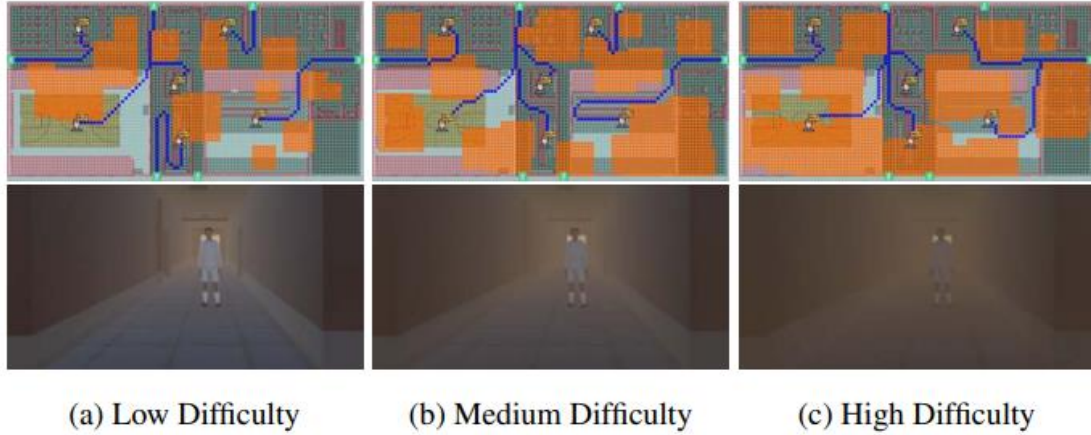


Figure 28: The three experimental conditions we used for our user study. Top: The position and size of fires (orange cells), the optimal paths (blue cells), and the position of the virtual agents. Bottom: The visibility of each training drill. © [2022] IEEE

4.6.3 Measurements

We collected participants’ perceived mutual awareness, mutual assistance, and dependent actions based on the questionnaire developed by Biocca et al. [4]. For each question, we used a 7-point Likert scale. In addition, we collected several in-game measurements to record participants’ collaborative behavior. These in-game measurements include the completion time, completion time offset, trajectory length, distance between participants, extinguisher counts, and number of commands.

4.6.4 Procedure

After we grouped the volunteers, we scheduled each group a specific time slot to attend the study at their corresponding university campus. Once both participants arrived, we first asked them to sign the consent form, which was approved by each university’s Institutional Review Board (IRB), if they agreed to participate. Next, the research team collected the demographic information from the participants by asking them to fill out a questionnaire. Then, our research team introduced and helped the participants with the experiment procedures and virtual reality equipment.

Participants first joined the warm-up session to meet in the warm-up scene; integrating a tutorial session improves participants’ performance and experience [14]. The warm-up scene was different from the experiment scenes, but all the interaction mechanisms were the same. We

instructed them to familiarize themselves with the voice commands and their functionality. Next, the research team informed them how to use the fire extinguisher and enable the minimap (see Figure 29a), and at the same time, they became familiar with the Virtuix Omni treadmill. Once participants finished the warm-up session and agreed to start the experiment, the research team helped them join the experiment’s scene. In Figure 29b, we show two users trying to open a path using the fire extinguisher. The warm-up session took no more than five minutes, and each experiment session lasted about 10 minutes (no participant spent more than one hour to complete the entire study). We informed participants they were allowed to give up the study; however, no participant quit.



Figure 29: (a) Users can enable a minimap. The minimap provided information on players’ position, the position of the virtual agents, the exits, and the commands they could use. (b) Two users collaborate in the shared virtual environment to open a path to escape the building.

© [2022] IEEE

4.6.5 Setup and Implementation Details

We used Unity Game Engine 2020.3.20f1 to develop the application. We also used a Dell Alienware Aurora R7 desktop computer (Intel Core i7, NVIDIA GeForce RTX 2080, 32GB RAM) in each university to run the application. We used Unity’s Photon asset to implement the network frame to allow participants to communicate and collaborate in a shared virtual space. The optimization process for authoring each training drill did not exceed 30 seconds. We used the Virtuix Omni treadmill to allow participants to move around in the virtual environment and Oculus Quest 2 as a VR headset. Lastly, we used the KeywordRecognizer class provided by Microsoft and integrated it into the UnityEngine library for voice recognition.

4.6.6 Results

We used one-way repeated measures analysis of variance (ANOVA) to analyze the data collected from three experimental conditions (LD, MD, and HD). We assessed the individual differences using post-hoc Bonferroni corrected estimates if the ANOVA was statistically significant. We provide the descriptive statistics in the supplementary materials file.

4.6.6.1 Objective Data

The analysis revealed a statistically significant result for the completion time measurement across the three examined conditions ($\Lambda = .413, F[2, 25] = 17.791, p = .000, \eta_p^2 = .587$). The post-hoc pairwise comparison showed that the completion time during the LD condition was significantly lower than that for the MD ($p = .030$) and HD ($p = .000$) conditions. Moreover, the completion time was significantly lower for the MD condition than the HD condition ($p = .012$). We also found a statistically significant result for the extinguisher count measurement ($\Lambda = .381, F[2, 52] = 42.179, p = .000, \eta_p^2 = .619$). The post-hoc pairwise comparison revealed that our participants used the virtual extinguisher less often in the LD condition than the MD ($p = .000$) and HD ($p = .000$) conditions; moreover, the participants used the virtual extinguisher less often during the MD condition than the HD condition ($p = .019$). However, the statistical analysis did not reveal significant results for the **completion time offset** ($\Lambda = .966, F[2, 25] = .441, p = .649, \eta_p^2 = .034$), **trajectory length** ($\Lambda = .942, F[2, 52] = 1.592, p = .213, \eta_p^2 = .058$), **distance between participants** ($\Lambda = .883, F[2, 25] = 1.663, p = .210, \eta_p^2 = .117$), and **number of commands** ($\Lambda = .962, F[2, 52] = 1.033, p = .363, \eta_p^2 = .038$). We provide the descriptive statistics in the supplementary materials file.

4.6.6.2 Subjective Self-reported Data

The **mutual awareness** measurement was statistically significant ($\Lambda = .618, F[2, 52] = 16.062, p = .000, \eta_p^2 = .382$) across the three examined conditions. The post-hoc pairwise comparison showed that mutual awareness was significantly lower during the LD condition than the MD ($p = .000$) and HD ($p = .000$) conditions. Similarly, mutual assistance was statistically significant ($\Lambda = .593, F[2, 52] = 17.877, p = .000, \eta_p^2 = .407$). The post-hoc pairwise

comparison revealed that mutual assistance was significantly lower during the LD condition than the MD ($p = .034$) and HD ($p = .000$) conditions, and the MD condition was significantly lower than the HD condition ($p = .001$). The dependent actions measurement was also statistically significant across the three conditions ($\Lambda = .286, F[2, 52] = 64.943, p = .000, \eta_p^2 = .717$). The post-hoc pairwise comparison showed that dependent actions were rated significantly lower during the LD condition than the MD ($p = .000$) and HD ($p = .000$) conditions, and the MD condition was rated significantly lower than the HD condition ($p = .000$).

4.6.7 Discussion

The collected objective data, and more specifically the completion time and extinguisher count measurements, revealed that our method can automatically synthesize training drills that have different difficulty levels for executing them. These findings prove that it is possible to synthesize fire evacuation training drills in which the trainer/designer can specify the parameters, such as the path length, number of turns in the optimal paths, number of fires, environment visibility, and the system can synthesize variations of the training drill without impacting the overall objective of that drill. However, the trajectory length measurement was not statistically significant across the three examined conditions. Considering that our participants walked the same trajectory lengths across the three conditions, the completion time proves that they needed more time to complete a more difficult training drill in comparison to the MD or LD training drills, in which they extinguish fewer fires and had higher visibility. If we also consider the number of commands measurement, we could say that our participants tried to instruct the virtual agents in roughly the same way across the three conditions. Thus, we can say that the virtual fires (due to completion time and extinguisher count) impacted our participants' behavior in executing the tasks, but not the virtual agents. Consequently, we argue that our method can synthesize training drills based on the difficulty entailed in executing them.

In contrast, the other measurements did not differ across the three experimental conditions. Specifically, an interesting observation was made for the completion time offset and the distance between participants measurements. In both measurements, although the completion time offset and the distance between participants decreased from the LD condition to the MD condition and from the MD condition to the HD condition, the decreases were not statistically significant. However, by looking at the mean values for the completion time offset measurement, it is evident

that the time offset is close to 30 seconds for all three conditions. A similar observation can be made for the distance between the participants: their mean distance is sufficient across the three conditions, which indicates that they were in different locations in the building during the training drills. These findings suggest that although the participants were in the same shared space, they chose their strategies and acted independently. Such independent activity has been identified by Tang et al. [28] as the “same problem, different area” style of coupling between two people. Therefore, we think that our two participants preferred to utilize collaborative behavior that could help them execute the given task in a way that was more optimal for them.

The mutual awareness measurement indicated that the participants were aware of each other during the training drill. It seems that the difficulty of the training drill impacted their awareness of one another. Therefore, the participants felt they were not alone while executing the given task in the virtual environment. The mutual assistance and dependent actions measurements revealed that, as the difficulty level of the training drill increased, the mutual assistance of each participant (the degree to which each person needed to help the other person) and their perceived dependence on the other participant increased. These findings indicate that the participants felt the pressure of the training drill, and they tried to assist the other participant by creating a strategy that would help them execute the given task and assist the other person.

Overall, by combining both the objective and self-reported measurements, we can say that, though our participants planned their strategy independently of each other, they were always aware of the other individual in the shared virtual environment, and given their awareness, they planned their strategy to help not only themselves but also the other participant. It looks as if this kind of planning is common in games [1] where players on the same team work together to accomplish a given task. Our results showed that, though the two participants were in separate locations, being in a shared virtual space and sharing the same goals and tasks made them choose individual strategies that benefited themselves and the team; therefore, establishing a collaborative culture.

4.6.8 Limitations

Our study had some limitations. First, our participants were not exposed to real-world evaluations. Therefore, we cannot firmly conclude that the training platform and its performance are effective in real-world emergency evacuation scenarios. Second, due to the hardware limitations (we used an Omni treadmill), long-time locomotive tasks will result in the users

needing to exert physical effort and experience fatigue [23], which could potentially decrease their motivation. Third, our optimization-based approach only considered four design decisions to synthesize the training drill. We think additional cost terms could be considered, such as those related to specific training objectives.

4.7 Conclusion

In this paper, we introduced a method to synthesize training drills for fire evacuation scenarios. Due to the proposed optimization-based formulation, a designer/trainer can easily define the target objectives for each cost term. Our system automatically synthesizes the training scenario where participants encounter the specified difficulty of executing a task. Thus, a designer/trainer could easily generate several variations of a training drill, allowing trainees to experience them and get prepared for potential real-world situations.

4.8 Acknowledgements

This project is partially supported by an NSF CAREER grant (award #: 1942531) and an NSF FW-HTF-R grant (award #: 2128867).

4.9 References

- [1] Alharthi, Sultan. A., Torres, Ruth. C., Khalaf, Ahmed. S., Toups, Zachary. O., Dolgov, Igor. and Nacke, Lennart. E. 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* (pp. 1-13).
- [2] Baker, Lindsay. 2012. A History of School Design and Its Indoor Environmental Standards, 1900 to Today. *National Clearinghouse for Educational Facilities*.
- [3] Barsom, E. Z., Graafland, M. and Schijven, M. P. 2016. Systematic review on the effectiveness of augmented reality applications in medical training. *Surgical endoscopy*, 30, 4174-4183.
- [4] Biocca, Frank., Harms, Chad. and Gregg, Jenn. 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 (pp. 1-9).
- [5] Burguillo, Juan. C. 2010. Using game theory and competition-based learning to stimulate student motivation and performance. *Computers & education*, 55(2), 566-575.

- [6] Chiang, Feng-Kuang., Shang, Xiaojing. and Qiao, Lu. 2022. Augmented reality in vocational training: A systematic review of research and applications. *Computers in Human Behavior*, 129, 107-125.
- [7] Chib, Siddhartha. and Greenberg, Edward. 1995. Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4), 327-335.
- [8] Dillenbourg, Pierre. 1999. Collaborative learning: Cognitive and computational approaches. advances in learning and instruction series. *Elsevier Science, Inc.*, PO Box 945, Madison Square Station, New York, NY 10160-0757.
- [9] Elvezio, Carmine., Sukan, Mengu., Oda, Ohan., Feiner, Steven. and Tversky, Barbara. 2017. Remote collaboration in AR and VR using virtual replicas. In *ACM SIGGRAPH 2017 VR Village* (pp. 1-2).
- [10] Freina, Laura. and Ott, Michela. 2015. A literature review on immersive virtual reality in education: state of the art and perspectives. In *The international scientific conference elearning and software for education* (Vol. 1, No. 13, pp. 10-1007).
- [11] Gajda, Rebecca. 2004. Utilizing collaboration theory to evaluate strategic alliances. *American journal of evaluation*, 25(1), 65-77.
- [12] Green, Peter. J. 1995. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4), 711-732.
- [13] Greenwald, Scott. W., Kulik, Alexander., Kunert, Andre., Beck, Stephan., Fröhlich, Bernd., Cobb, Sue., Parsons, Sarah., Newbutt, Nigel., Gouveia, Christine., Cook, Claire., Snyder, Anne., Payne, Scott., Holland, Jennifer., Buessing, Shawn., Fields, Gabriel., Corning, Wiley., Lee, Victoria., Xia, Lei. and Maes, Pattie. 2017. Technology and applications for collaborative learning in virtual reality. *Philadelphia, PA: International Society of the Learning Sciences*.
- [14] Kao, Dominic., Magana, Alejandra. J. and Mousas, Christos. 2021. Evaluating tutorial-based instructions for controllers in virtual reality games. *Proceedings of the ACM on Human-Computer Interaction*, 5(CHI PLAY), 1-28.
- [15] Kaplan, Alexandra. D., Cruit, Jessica., Endsley, Mica., Beers, Suzanne. M., Sawyer, Ben. D. and Hancock, P. A. 2021. The effects of virtual reality, augmented reality, and mixed reality as training enhancement methods: A meta-analysis. *Human factors*, 63(4), 706-726.
- [16] Kurillo, Gregorij., Bajcsy, Ruzena., Nahrsted, Klara. and Kreylos, Oliver. 2008. Immersive 3d environment for remote collaboration and training of physical activities. In *2008 IEEE Virtual Reality Conference*. pp. 269-270. IEEE.
- [17] Lai, Emily. R. 2011. Collaboration: A literature review. *Pearson Publisher*, vol. 11, 2016.

- [18] Lee, Kangdon. 2012. Augmented reality in education and training. *TechTrends*, 56(2), 13.
- [19] Li, Changyang., Liang, Wei., Quigley, Chris., Zhao, Yibiao. and Yu, Lai-Fai. 2017. Earthquake safety training through virtual drills. *IEEE transactions on visualization and computer graphics*, 23(4), 1275-1284.
- [20] Liu, Huimin., Wang, Zhiquan., Mousas, Christos. and Kao, Dominic. 2020. Virtual reality racket sports: Virtual drills for exercise and training. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. pp. 566-576. IEEE.
- [21] Luxenburger, Andreas., Prange, Alexander., Moniri, Mohammad. M. and Sonntag, Daniel. 2016. Medicalvr: Towards medical remote collaboration using virtual reality. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. pp. 321-324.
- [22] Martins, Bruna. R., Jorge, Joaquim. A. and Zorzal, Ezequiel. R. (2021). Towards augmented reality for corporate training. *Interactive Learning Environments*, 1-19.
- [23] Mousas, Christos., Kao, Dominic., Koiliias, Alexandros. and Rekabdar, Banafsheh. 2021. Evaluating virtual reality locomotion interfaces on collision avoidance task with a virtual character. *The Visual Computer*, 37, 2823-2839.
- [24] Piumsomboon, Thammathip., Lee, Gun. A. and Billingham, Mark. 2018. Snow dome: A multi-scale interaction in mixed reality remote collaboration. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. pp. 1-4.
- [25] Piumsomboon, Thammathip., Lee, Gun. A., Hart, Jonathon. D., Ens, Barrett., Lindeman, Robert. W., Thomas, Bruce. H. and Billingham, Mark. 2018. Mini-me: An adaptive avatar for mixed reality remote collaboration. In *Proceedings of the 2018 CHI conference on human factors in computing systems* (pp. 1-13).
- [26] Roschelle, Jeremy. and Teasley, Stephanie. D. 1995. The construction of shared knowledge in collaborative problem solving. In *Computer supported collaborative learning* (pp. 69-97). Springer Berlin Heidelberg.
- [27] Ruikar, Darshan. D., Hegadi, Ravindra. S. and Santosh, K. C. 2018. A systematic review on orthopedic simulators for psycho-motor skill and surgical procedure training. *Journal of medical systems*, 42, 1-21.
- [28] Tang, Anthony., Tory, Melanie., Po, Barry., Neumann, Petra. and Carpendale, Sheelagh. 2006. Collaborative coupling over tabletop displays. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (pp. 1181-1190).

- [29] Tea, Sunseng., Panuwatwanich, Kriengsak., Ruthankoon, Rathavoot. and Kaewmoracharoen, Manop. 2022. Multiuser immersive virtual reality application for real-time remote collaboration to enhance design review process in the social distancing era. *Journal of Engineering, Design and Technology*, 20(1), 281-298.
- [30] Viglialoro, Rosanna. M., Condino, Sara., Turini, Giuseppe., Carbone, Marina., Ferrari, Vincenzo. and Gesi, Macro. 2021. Augmented reality, mixed reality, and hybrid approach in healthcare simulation: a systematic review. *Applied Sciences*, 11(5), 2338.
- [31] Villanueva, Ana., Zhu, Zhengzhe., Liu, Ziyi., Wang, Feiyang., Chidambaram, Subramanian., and Ramani, Karthik. 2022. Colabar: A toolkit for remote collaboration in tangible augmented reality laboratories. *Proceedings of the ACM on Human-Computer Interaction*, 6(CSCW1), 1-22.
- [32] Webel, Sabine., Bockholt, Uli., Engelke, Timo., Gavish, Nirit., Olbrich, Manuel. and Preusche, Carsten. 2013. An augmented reality training platform for assembly and maintenance skills. *Robotics and autonomous systems*, 61(4), 398-403.
- [33] Williams, E. J. 1949. Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Chemistry*, 2(2), 149-168.
- [34] Xie, Biao., Liu, Huimin., Alghofaili, Rawan., Zhang, Yongqi., Jiang, Yeling., Lobo, Flavio. Destri., Li, Changyang., Li, Wanwan., Huang, Haikun., Akdere, Mesut., Mousas, Christos. and Yu, Lap-Fai. 2021. A review on virtual reality skill training applications. *Frontiers in Virtual Reality*, 2, 645153.
- [35] Yao, Junfeng., Lin, Chao., Xie, Xiaobiao., Wang, Andy. J. and Hung, Chih-Cheng. 2010. Path planning for virtual human motion using improved A* star algorithm. In *2010 Seventh international conference on information technology: new generations* (pp. 1154-1158). IEEE.

DISCUSSION AND CONCLUSION

This dissertation aimed to develop a framework for formulating computational design problems using Markov chain Monte Carlo (MCMC) optimization theory. The framework enabled designers to encode their design considerations as cost terms and solve optimization-based problems using computational methods to generate virtual contents. Three research questions were addressed, namely (1) the feasibility of formulating design problems using optimization theory, (2) suitable/viable application cases, and (3) suitable/viable cost terms for each application case based on the theory.

This dissertation contains the papers published by the author during her Ph.D. Each published article included in this dissertation deals with a specific application case based on optimization theory.

Four application cases were explored, each supported by a published article included as an individual chapter of the dissertation. The first application case involved the design of virtual reality racket sports drills as an optimization problem. The second case focused on virtual reality game level layout design with real environment constraints. The third case addressed the design of collaborative gameplay in a shared virtual environment. The fourth application case involved synthesizing shared space virtual reality fire evacuation training drills. The definition of cost terms varied for each scenario, and designers combined specific scenario domain knowledge to define cost terms that successfully generated the objective scenario.

The resulting synthesis was successful, producing different synthesized results based on different target cost input values and weights according to the theory. This positively answered the first research question. The synthesized results triggered statistically significant differences in human behavior, demonstrating the validity of the formulation and answering the second research question. The dissertation also explored and discussed different cost terms based on various scenarios in each article, answering the third research question.

APPENDIX

A 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:** Leaves 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.

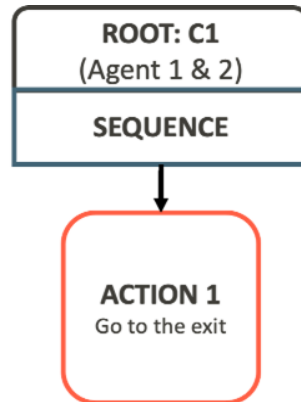


Figure 30: Behavior tree for the C1 game level chunk (Nodes: 2; Depth: 1)

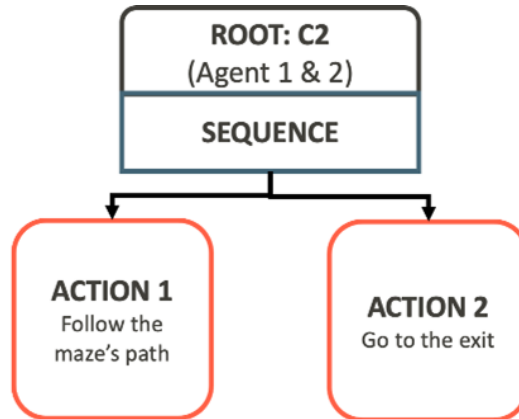


Figure 31: Behavior tree for the C2 game level chunk (Nodes: 3; Depth: 1)

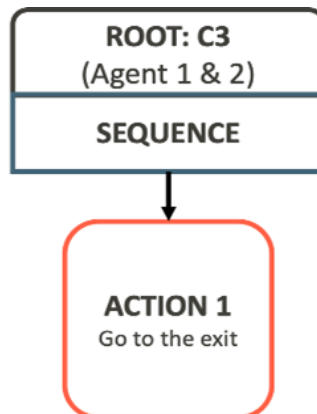


Figure 32: Behavior tree for the C3 game level chunk (Nodes: 2; Depth: 1)

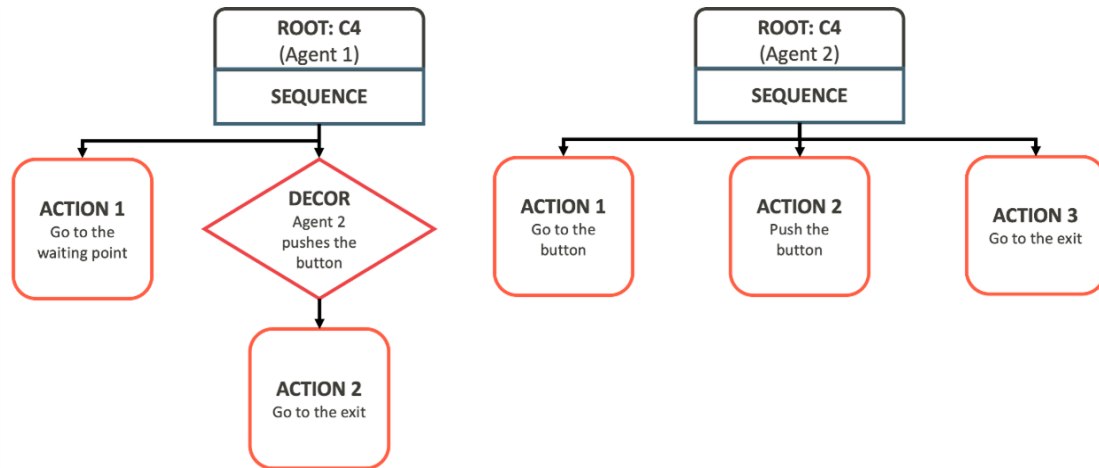


Figure 33: Behavior trees for the C4 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 4; Depth: 1] ;)

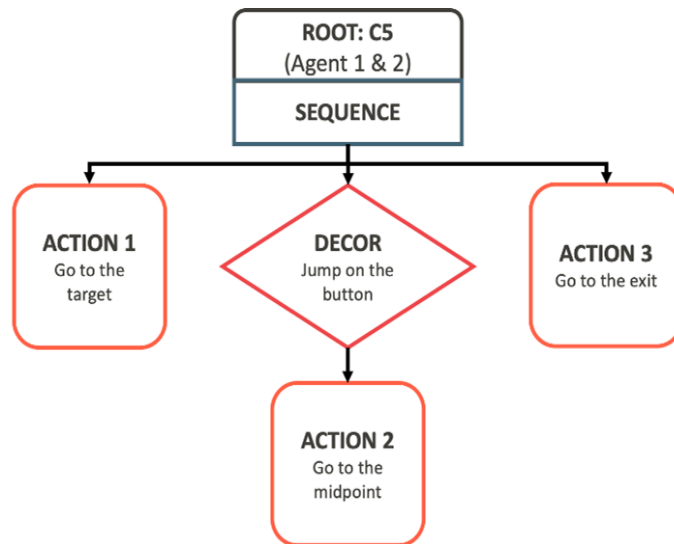


Figure 34: Behavior tree for the C5 game level chunk (Nodes: 5; Depth: 2)

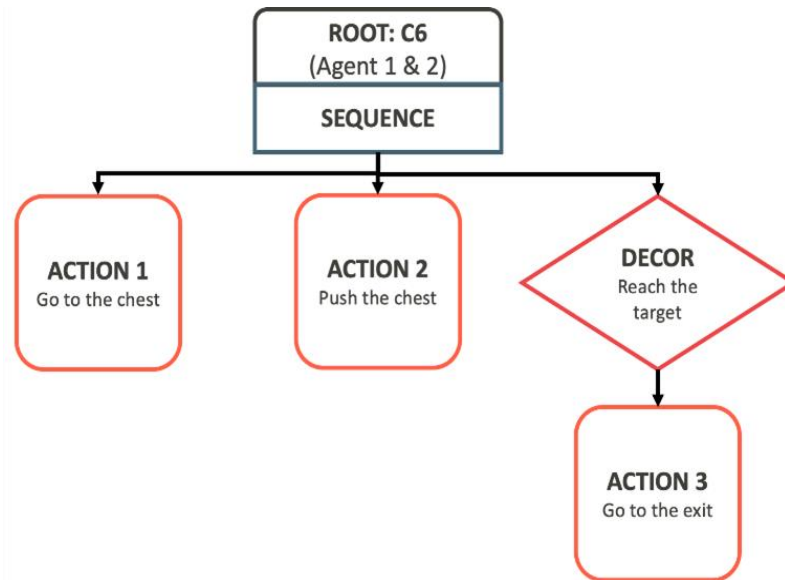


Figure 35: Behavior tree for the C6 game level chunk (Nodes: 5; Depth: 2)

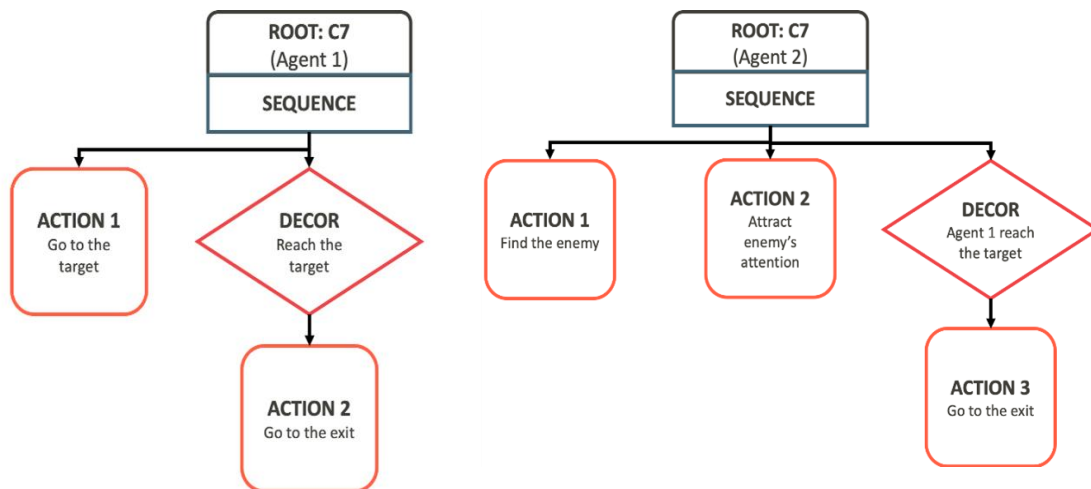


Figure 36: Behavior trees for the C7 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2] ;)

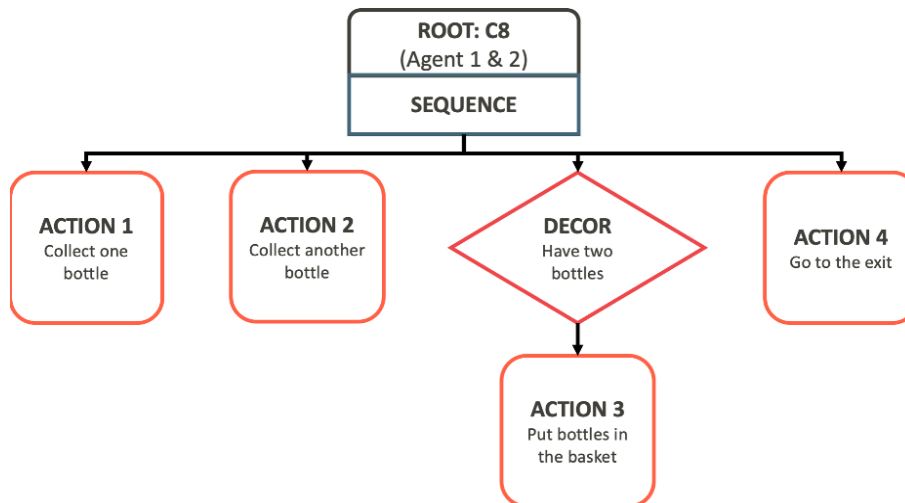


Figure 37: Behavior tree for the C8 game level chunk (Nodes: 6; Depth: 2)

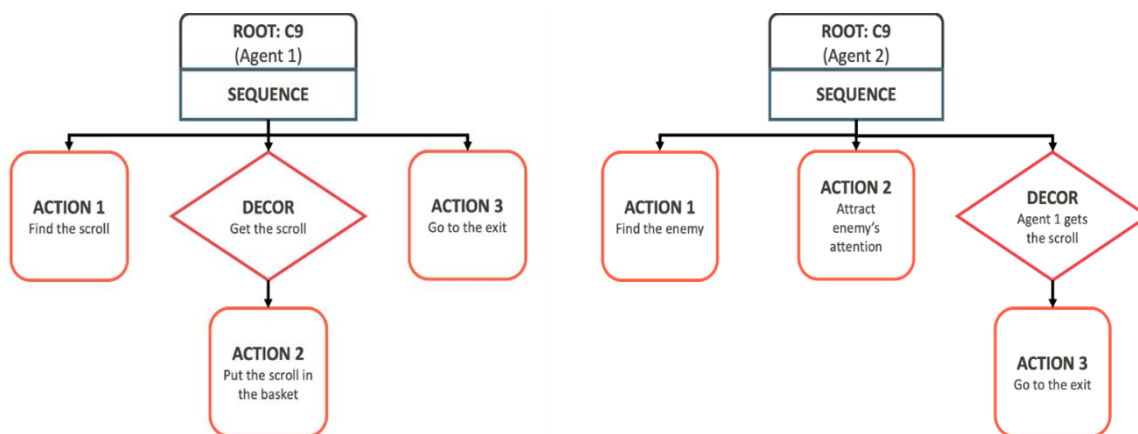


Figure 38: Behavior trees for the C9 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2] ;)

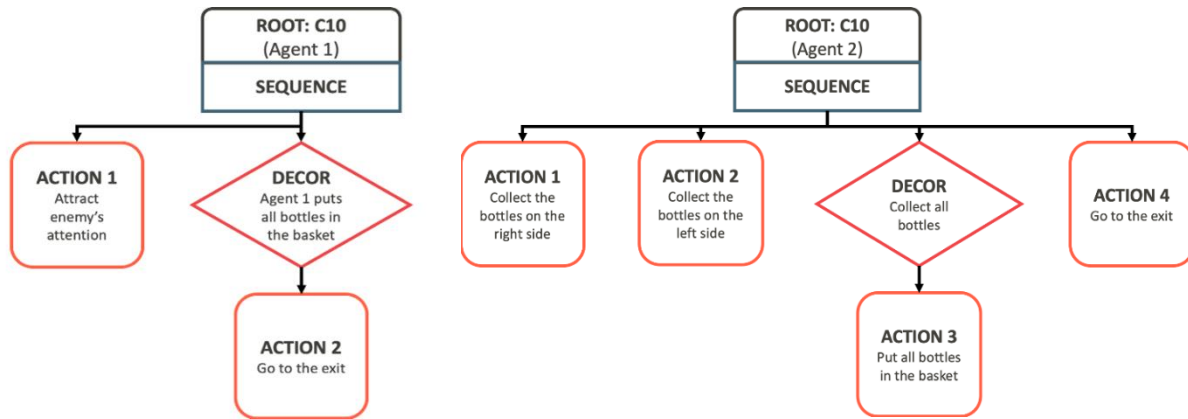


Figure 39: Behavior trees for the C10 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 6; Depth: 2] ;)

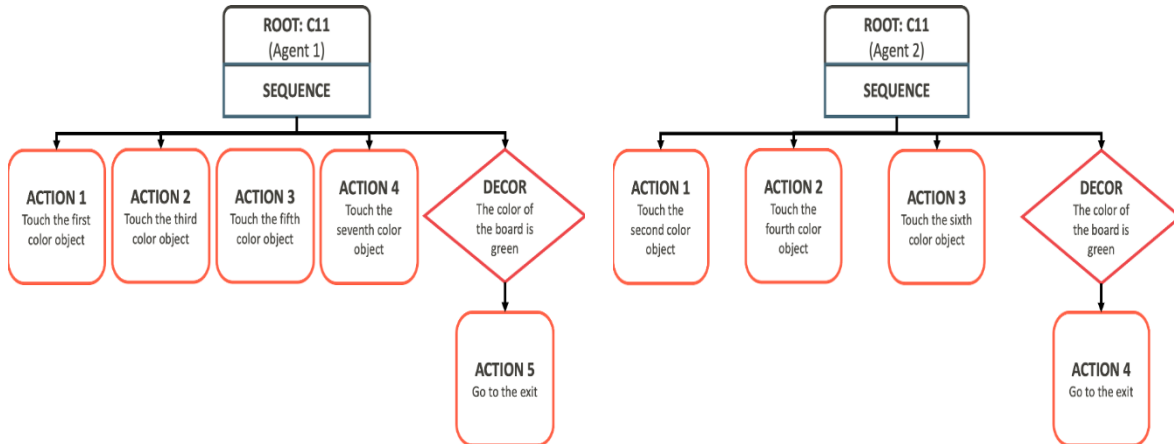


Figure 40: Behavior trees for the C11 game level chunk (Left: Player 1 [Nodes: 7; Depth: 2]; Right: Player 2 [Nodes: 6; Depth: 2] ;)

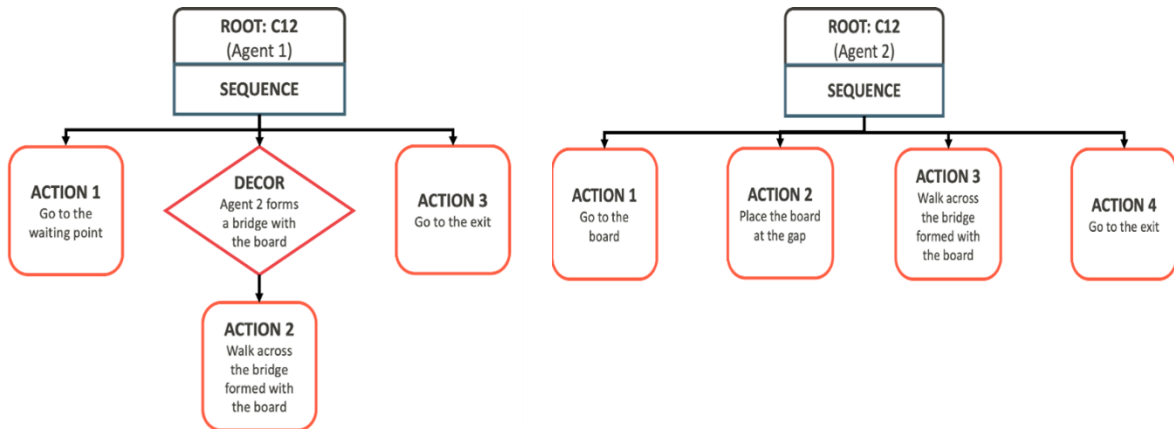


Figure 41: Behavior trees for the C12 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 1] ;)

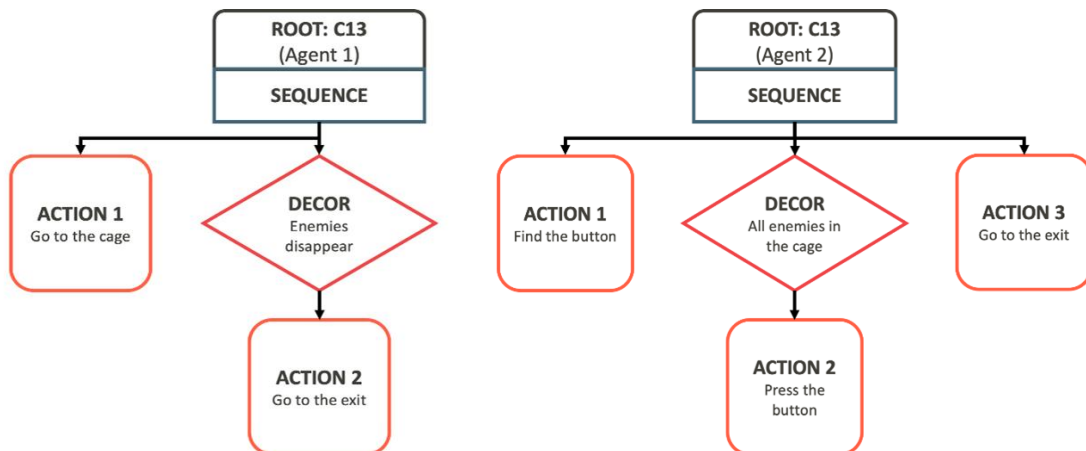


Figure 42: Behavior trees for the C13 game level chunk (Left: Player 1 [Nodes: 4; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2] ;)

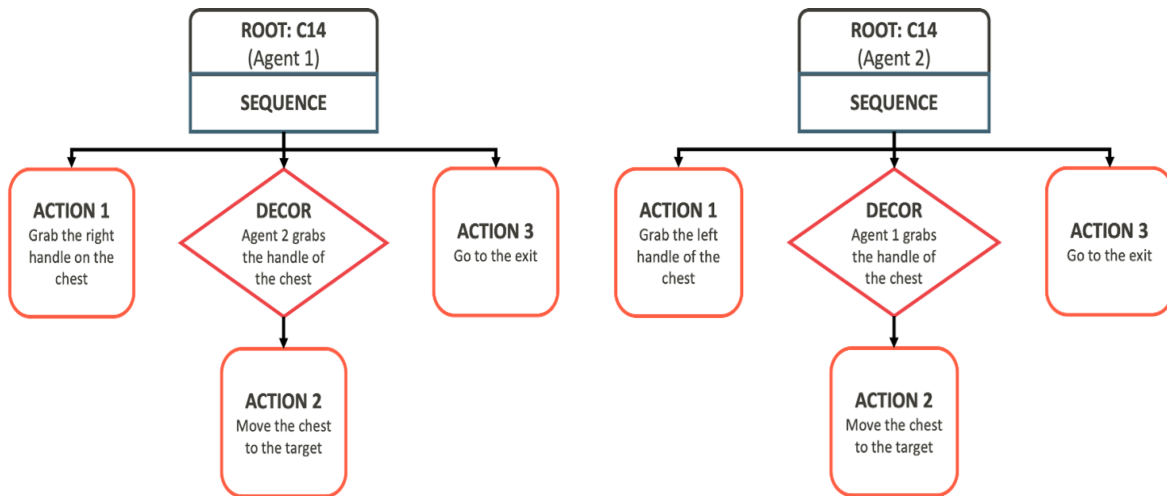


Figure 43: Behavior trees for the C14 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 5; Depth: 2] ;)

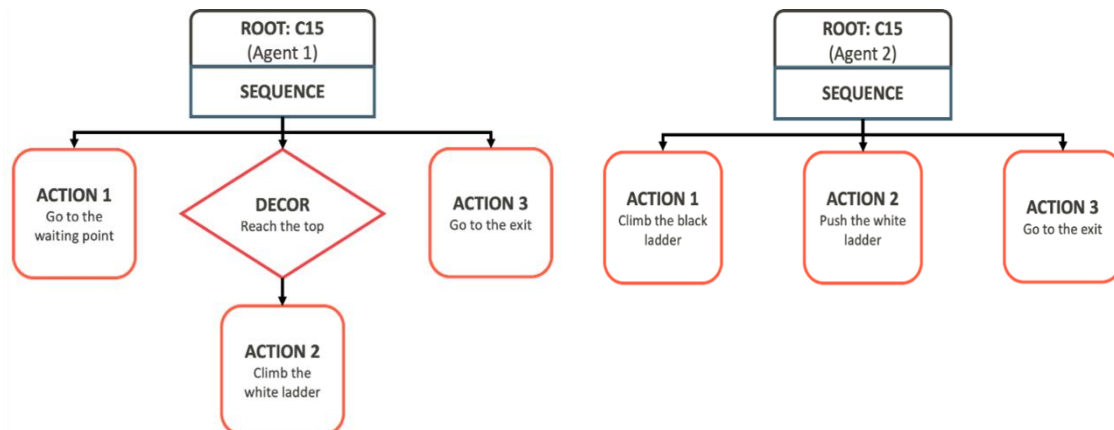


Figure 44: Behavior trees for the C15 game level chunk (Left: Player 1 [Nodes: 5; Depth: 2]; Right: Player 2 [Nodes: 4; Depth: 1] ;)

PUBLISHED ARTICLES

- **Liu, Huimin.**, Wang, Zhiquan., Mousas, Christos. and Kao, Dominic. 2020. Virtual reality racket sports: Virtual drills for exercise and training. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (pp. 566-576). IEEE.
- **Liu, Huimin.**, Wang, Zhiquan., Mazumdar, Angshuman. and Mousas, C. 2021. Virtual reality game level layout design for real environment constraints. *Graphics and Visual Computing*, 4, 200020.
- **Liu, Huimin.**, Choi, Minsoo., Kao, Dominic. and Mousas, Christos. 2022. Synthesizing Game Levels for Collaborative Gameplay in a Shared Virtual Environment. *ACM Transactions on Interactive Intelligent Systems (TiiS)*.
- **Liu, Huimin.**, Choi, Minsoo., Yu, Liuchuan., Koiliias, Alexandros., Yu, Lap-Fai. and Mousas, Christos. 2022. Synthesizing Shared Space Virtual Reality Fire Evacuation Training Drills. In *2022 IEEE International Symposium on Mixed and Augmented Reality (Adjunct) (ISMAR Adjunct)* (pp. 459-464). IEEE.
- Wang, Zhiquan., **Liu, Huimin.**, Pan, Yucong. and Mousas, Christos. 2020. Color blindness bartender: An embodied VR game experience. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)* (pp. 519-520). IEEE.
- Xie, Biao., **Liu, Huimin.**, Alghofaili, Rawan., Zhang, Yongqi., Jiang, Yeling., Lobo, Flavio. Destri., Li, Changyang., Li Wanwan., Huang, Haikun., Akdere, Mesut., Mousas, Christos. and Yu, Lap-Fai. 2021. A review on virtual reality skill training applications. *Frontiers in Virtual Reality*, 2, 645153.
- Acevedo, Pedro., Choi, Minsoo., **Liu, Huimin.**, Kao, Dominic. and Mousas, Christos. 2022. Procedural Game Level Design to Trigger Spatial Exploration. In *Proceedings of the 17th International Conference on the Foundations of Digital Games* (pp. 1-11).