MULTIPLAYER GAME FRAMEWORKS FOR CROWD-AWARE CO-DESIGN OF ENVIRONMENTS

BY NILAY CHAKRABORTY

A thesis submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Master of Science

Graduate Program in Computer Science

Written under the direction of

Dr. Mubbasir Kapadia

and approved by

____

____

____

____

New Brunswick, New Jersey

May, 2017

ABSTRACT OF THE THESIS

Multiplayer Game Frameworks for Crowd-Aware Co-design of Environments

by Nilay Chakraborty

Thesis Director: Dr. Mubbasir Kapadia

This paper proposes two frameworks in the field of Games in Research.

- **Proposal 1**

  Architectural design decisions stand to benefit by accounting for the presence and activities of human crowds that inhabit these spaces. Computational methods for simulating synthetic crowds provide a cost-effective means of exploring and analysing the design search space. However, the crowd-aware architectural design is a complicated combinatorial decision process, where small changes in the design solution may affect crowds and their flow patterns in unexpected and potentially unintuitive ways. Existing solutions rely on expert intuition or automation, but are unable to account for much contradicting, and often subjective properties while optimising designs. A single solution approach may also miss potential design solutions that achieve the desired objective specifications while meeting subjective criteria. We propose a means of combining these methods in a gamified framework. Using our system, “players" (novice users or experts) can rapidly iterate on their designs while soliciting feedback from computer simulations of crowd movement, as well as the designs of previous players. Our approach affords a new way of thinking of the solution space in that it inherently supports competitive co-design and crowdsourced solutions. We evaluate our framework through a user study and demonstrate the potential

of crowd-aware co-design of environments using simulation guided multiplayer games.

- **Proposal 2**

  Games in research is a modern trend in the world of research. Games are being used to simulate, monitor and test occurrences in reality. However, Game development knowledge is the initial requirement to use games in research either in any non-technical or technical field. Besides, availability of existing games suited to research is very meager. Also, attracting and reaching out to enough users for proper user study is a difficult task. We propose a connected real-time website and desktop-based game application. The website is themed as a social network which will help connect researchers with gamers and keep them updated. Our application also allows a researcher to choose and modify a few configuration parameters to change the game according to his requirement as well as dedicated ownership to the collected data. The game application provides a robust Massive Multiple Online game environment for a smooth and exciting gaming experience to involve gamers to contribute to research.

Acknowledgements

I would like to thank my adviser, Prof. Mubbasir Kapadia, without whose help and guidance I would not have achieved this.

Dedication

This work is dedicated to everyone who supported me through this endeavor.

Table of Contents

Chapter 1

Introduction

1.1 Contribution 1

Designing an architectural or urban plan is a complex problem in which a good solution is one that balances aesthetics, functionality, utilisation, and safety of a structure. This makes the problem inherently multi-objective and the solution space combinatorial. Incorporating the dynamics of large groups of people, or crowds, further complicate the solution space and are prohibitively expensive to do with real people. Thus layout designs are commonly tested using synthetic crowds that model realistic behaviours under different conditions. In particular, a layout's performance is most critical during dangerous situations such as evacuations, so these scenarios are used to stress test environments.

Popular architectural design tools such as Autodesk Revit and Rhino3D do not account for evacuation planning or crowd-aware stress testing. There are tools used to analyse a design with simulated crowds, but none of that tool has an integrated environment for both architectural design and crowd simulation. Together these tools are quite complex an require years of training and architectural and safety knowledge.

Fully automated computational solutions for crowd-aware architectural design may miss solutions which are technically sound, crowd-aware, and aesthetically pleasing. These rely heavily on the quality of the objective functions for each of these requirements. Recent work in this field has sought for user-in-the-loop optimisation processes which make up for these shortcomings and provide the user more control over solution directions.

Our solution is to gamify the process of optimal layout design for crowd-aware architectures, or environments. This game provides the player with feedback regarding crowd simulation and heatmaps of evacuation dynamics while affording them editing power within the constraints imposed by the architect or designer. The approach of gamifying complicated problems affords many benefits to both the process and the solution. The most immediate are that the game provides a fun and interactive platform for architectural and urban planning.

By providing a means to implement the design process as gamified levels with built-in architectural constraints, a planner, environment designer, or architect can convert their design problem into a playable game. This re-imagines the architect or planner seeking a design solution as a game maker.

Furthermore, providing the game via the Internet, or large scale network of collaborators, design solutions may be crowdsourced. A user with any level of knowledge in a related field can produce an optimal design plan. This can also be used as an exciting learning platform in the field of architectural education.

Providing the results of other players designs and their fitness affords a further level of collaboration in the form of competitive co-design. The collective knowledge of the crowd, as in collaborators, and iterative improvement of co-design will help the layout to converge towards optimality.

1.2 Contribution 2

Video games are nowadays making the boundaries between virtual and real world disappear by using incredibly realistic graphics and exhaustive open-world environments. Games have now surpassed the purpose of entertainment and actively involved in the world of research. Games in research is a modern trend. To test and gather information about real world occurrences which can or cannot be mimicked by people, games are being used as a virtual substitute of the real world environments and events. Researchers at present surveying and monitoring human behaviour in a virtual world event. However to test a real world event the basic need is the technical skills to develop a game. Even if with skilful game developers, another need remains is to make the game available to the gamers because a successful research strongly needs a proper user study of enough user data to correctly hypothesise.

Currently, there are few game arcades which provides the gameplay datasets to researchers. Most of the time the researcher finds it challenging and exhaustive to search for a game which will meet their hypothesis requirement. Even if they do not find a game entirely suitable to their research needs, few small changes in game configuration can make the game meet their needs. However, the game arcades do not have the option to change few configuration for the game to meet their requirement.

In order to perform a better research using a game, a multiplayer facility in a game is adequate for the study because through this multiplayer platform a researcher can monitor and collect data of different types of user behaviour at the same time. This also helps to monitor the interaction of real-world users in a real-world event which is simulated in the game. Although, online multiplayer games are very popular, to set up an online multiplayer environment requires a researcher to have strong technical skills.

We propose a connected real-time system which can meet the above requirements. Our system is a combination of a real-time social networking website and a Massive Multiplayer Online application framework.

First of all, the main aim to develop a social networking website is to bring together researchers and players in a single platform. Through this approach, a researcher will be able to reach out to the gamers to gather enough data for successful research.

Secondly, our system provides a robust, massively multiplayer online environment with a set of exciting games to handle large traffic effectively to provide gamers with an exciting and smooth gaming experience. The games tasks are simple enough to meet a research requirement.

Also, the framework contains a set of configuration parameter and a set of values for each parameter, defined for each game. For each combination of these parameters, a different game can be formed. This capability will help a researcher to be able to make a game much more suitable according to the research requirement. To provide data privacy, we have provided dedicated solo ownership of the game user data to the researcher who has created it.

Chapter 2

Related Work

- **Contribution 1**

  There is an established and growing interest in the use of architectural optimization to explore design spaces and provide optimal solutions with respect to problem criteria [11, 49]. While there are popular CAD solution tools, such as Autodesk Revit and Rhino3D, these tools do not account for crowd behaviour and require significant training and background knowledge. Architectural optimization solutions generate new layouts or topology for structures with respect to objective and/or subjective criteria, while providing a trade-off between automation and author precision. Data-driven approaches [38] learn layout configurations from a database of prior architecture design constrained to a particular design space (e.g., residential homes). In another approach Design objectives can be modeled as forces applied to a physical feature to generate layout designs automatically [3].

  There are a lot of existing research on crowd simulation. The Social Force model allows for highly representative models of human behaviour in simple situations that elicit reflexive reactions [37]. The most closely related to our work is the optimization of egress environments using crowds [28, 27]. This work does not incorporate the user-in-the-loop processes and is presented as an analysis of the affects of egress obstacles and shapes on crowd evacuation. Since subjective criteria are difficult or impossible to quantify, many tools select an optimization scheme to meet objective criteria and then take a human-in-the-loop interactive approach to the subjective. The tools combine the aforementioned derivations for fitness with user-guided optimization processes [55, 59, 39].

  The maturity of research in simulating crowd dynamics [47] has resulted in a wide variety of approaches including social forces [23] and predictive models [60, 57]. There has been a growing recent trend to use statistical analysis in the evaluation and analysis of dense crowd simulations. The work in [19, 48] measures the ability of a steering algorithm to emulate the behavior of a real crowd data set by measuring its divergence from ground truth. Crowd optimization techniques [62, 8] automatically fit the parameters of a crowd to meet different criteria or even match real crowds.

We leverage this wealth of research in crowd simulation to present a computational design tool for environments that uses features extracted from an underlying crowd simulator as criteria for architectural optimization.

There is a plethora of research on crowd simulation and crowd evacuation using serious games. Most closest to our approach is [50] where serious games are used to training, planning and evaluating emergency plans but the research more focuses on crowd simulation component than architectural planning. Our Game focuses on optimizing the architectural design keeping in consideration the emergency situation using best suited simulated crowd. So that we can prevent a disaster at an earlier stage. Our game also considers the research paper [7] for characterizing and optimizing game level difficulty for our architectural planning game.

- **Contribution 2**

  Empirical analysis of crowd behavior though variations in behavior or environment dynamics is a modern trend in the field of games in research. Real world human behavior is monitored and modeled through game play data with varying environment. Specifically, an emergency situation which cannot be replicated, are simulated in a game environment and reactions of real human from a sudden predetermined set of rapid changes in the environment, are monitored and modeled through game play data. Some popular research in the related field is cited below.

  Some of notable researches using serious games include In a related research paper [24] a simple set of rules are defined which acts as variants for pedestrian behavior and which has high influence in causing a crowd disaster. This model predicts individual trajectories and collective patterns of motion in good quantitative agreement with a large variety of empirical and experimental data. It also predicts the emergence of self-organization phenomena, such as the spontaneous formation of unidirectional lanes or stop-and-go waves.

In another simulated crowd based experiment [16], The authors investigated the dynamics of steering and obstacle avoidance, with the aim of predicting routes through complex scenes. Participants walked in a virtual environment toward a goal and around an obstacle whose initial angle and distance varied. Goals and obstacles behave as attractors and repellers are variants here. The observed behavior was modeled as a dynamical system in which angular acceleration is a function of variants goal and obstacle angle and distance. By linearly combining terms for goals and obstacles, one could predict whether participants adopt a route to the left or right of an obstacle to reach a goal.

The paper on Simulating dynamical features of escape panic [15]though variation of individual pedestrian behavior they are trying to investigate the mechanisms of (and preconditions for) panic and jamming by uncoordinated motion in crowds in case of a panic situation.

Another important research paper [46] based on empirical analysis of crowd behavior where they present an evacuation experiment with over 500 participants testing individual behaviour in an interactive virtual environment. The influencing parameters are three different types of directional information: static information (signs), dynamic information (movement of the simulated crowd) and memorized information, as well as the combined effect of these different sources of directional information. In contrast to signs, crowd movement and learned information did not have a significant impact on human exit route choice in isolation.

Chapter 3

Framework Overview

3.1 Contribution 1

The game design gives players the tools to construct, modify, and analyse environment designs. A player can add elements to the provided floor design, such as doors, walls, and pillars. The player's goal and success criteria are to create an efficient design that evacuates all agents in the shortest possible time. The player can visualise the simulation and analyse the aggregate crowd dynamics using heat maps to identify regions for improvement. The player can use these metrics to improve the current design. By repeating a level and improving the evacuation time, the level layout can converge to an optimal design.

The following list provides an overview of the modules in the framework. The modules and their relationships are also outlined in Figure 3.1.

- **Static Initial Layout** - An architect or designer can designate static elements of the gamified environment - partially imposing architectural constraints. The player can not modify these elements in the existing layout. Users will have to place the elements provided to this layout. It is possible for the architect or designer to design each subsequent level such that the environment has increased the complexity of the existing design or is adds portions to the design. This means a player may work in a larger environment in which they can modify a portion, or the player may work from the micro to macroscopic portions of a larger design task.

- **Player Modified Layout** - There are certain elements available to the user to add to the layout. There are several elements which may be added or modified such as walls, pillars, and doors. Access to these modification and creation tools may be constrained by the architect or designer.

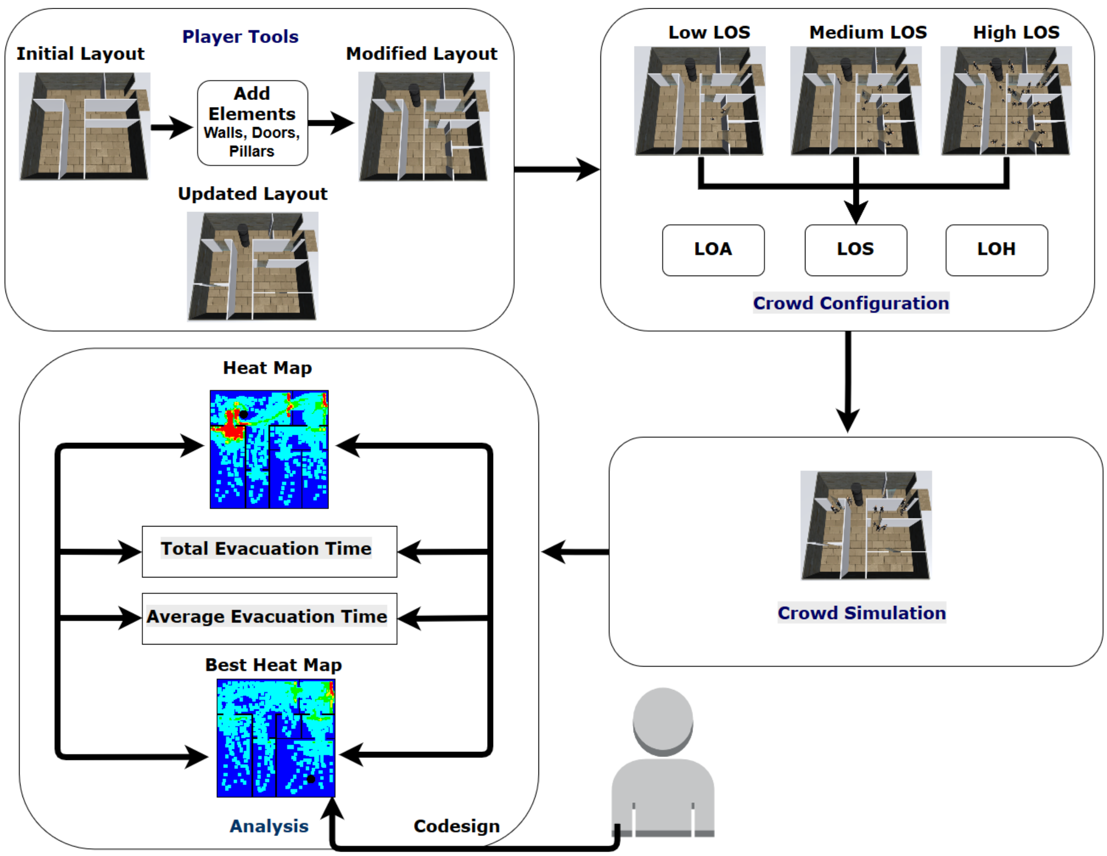


Figure 3.1: Architectural Planning and Crowd Evacuation Framework overview

- **Crowd Behaviour Configuration** - The player can configure the different Characteristics of the crowd to capture new behaviours and dynamics for which the environments fitness is tested regarding evacuation. A summary of these crowd parameters are listed below; in-depth details are provided in the Crowd Configuration and Simulation Section -

  1. **Level of Service (LoS)** - The number of people in a square unit of area that spawn when the level starts, or crowd density.
  2. **Level of Aggression** - The distribution of crowd member speed and acceleration.
  3. **Level of Homogeneity** - The distribution of crowd member width, or size.

  These parameters provide players with up to 54 unique combinations of crowd behaviour and dynamics.

- **Crowd Simulation** - Once the player finalises their environment configurations, they can run the simulation. The crowd will begin evacuating the layout towards exits defined by the architect or designer. An agent-based model using Unity3D is used for agent steering. The Unity 3D Mechanim system handles character animation, and the Navmesh system handles pathfinding. Once the entire crowd reaches their destination, the simulation ends.

- **Validation** - Once the simulation ends, the game displays the score based on the total evacuation time. The winning criteria, lower total evacuation time, determines where on the leader board the player is placed. In this phase, the player can visualise and analyse their results, regarding crowd dynamics, using heat maps.

- **Co-Design** - Players are given the ability to view other players designs and their performances. They may use various analytics to improve designs from a high scoring player. Co-design allows the design task to be tackled from multiple perspectives while converging on optimal designs.

3.2 Contribution 2

This massively multiplayer online(MMO) game system allows researchers the capability to choose, modify, test and collect data through a multiplayer game environment. The framework

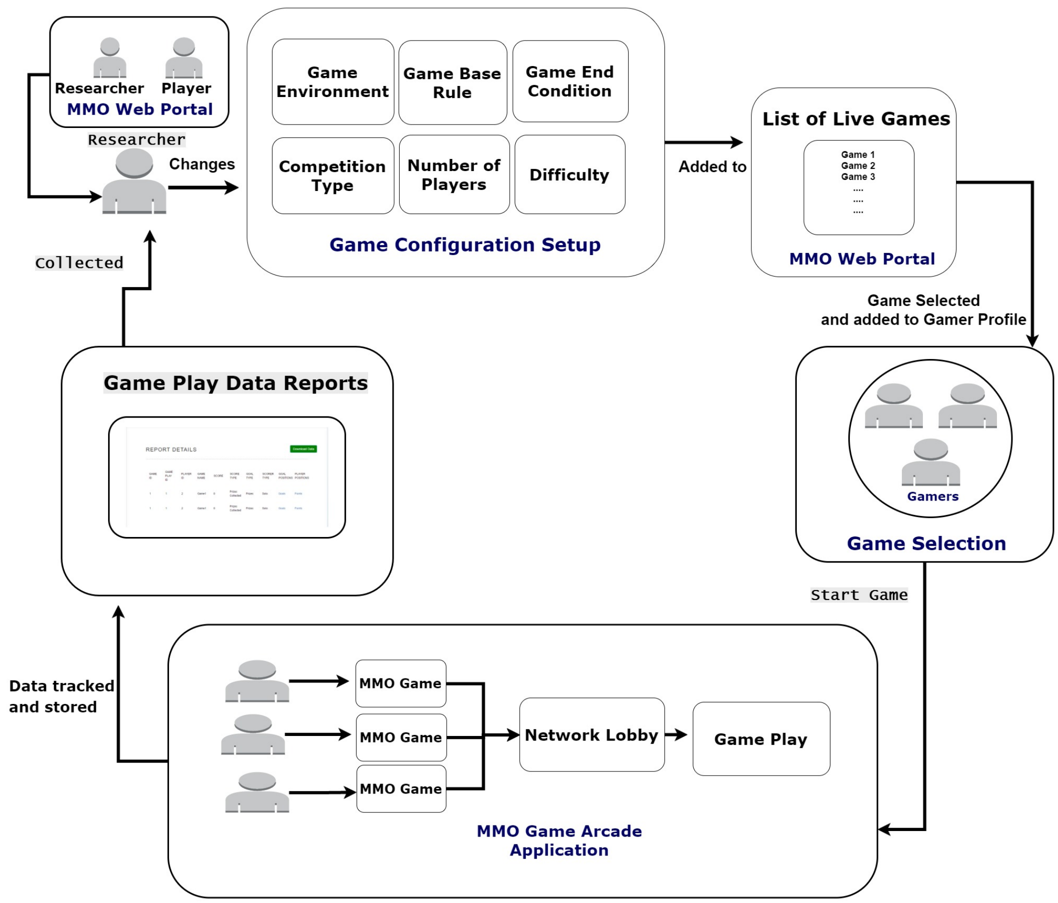


Figure 3.2: MMO Game Framework overview

also provides a robust MMO environment for a smooth and exciting multiplayer gaming experience for the gamers. A researcher can modify the values of the configuration parameters of the game to make the game closely provide the data required for the user study. The researcher can also choose from our list of data that can be tracked from the games, the data the researcher needs from this game. The researcher can add the game to the website. A player can choose a game from the website and add to the player profile. The player using the MMO application can play the games added to the user profile. Multiple players are come together in our robust multiplayer environment and play the game. Based on the researchers chosen options for data tracking, the application tracks those of those types from the game and provides to the researcher. The data is safely maintained, and the ownership is only assigned to the researcher.

The following list provides an overview of the modules in the framework. The modules and their relationships are also outlined in Figure 3.2.

- **MMO Web portal** - This is one of important building blocks of our system. MMO web portal is a social networking website where there are two types of users i)Researcher and ii)Gamer. The user privacy and user data security are maintained with user authentication. Both users have a live feed which shows updates of all the researcher as a default and also shows the updates of other gamers who are followed by the current user. The portal also has a capability of people search to search other gamers. A user can follow another user and get the followed user's updates in his/her live feed.

- **Game Configuration** - In the web portal, researchers can choose a game from the list of games in the portal and change the configurations of the games to make the game closer to the research requirement. The list of configurations supported by our framework is listed below.

  - **Game environment**

    This configuration controls the background location of the game.

  - **Game Base Rule**

    The basic rule of the game which is an implementation of the set of tasks a gamer has to perform which are different for each game. In our system, we have included four base games which can further be modified based on the configuration values.

- **Game End Strategy**

  This is the parameter within the game who's state is tracked to decide the occurrence of the end of the game. Our system supports to type of game end strategy as follows.

  1. Time completion
  2. Task completion

- **Game Difficulty**

  This is the game parameter which controls the difficulty in completion of a task within the game. Game difficulty changes the behaviour in different ways for different games. It can change the number of obstacles or the intelligence of the obstacles or time limitation etc. Following are the list of difficulties supported by our system.

  1. Easy
  2. Medium
  3. Hard

- **Game Competition Type**

  This is the game parameter which type of the game. This parameter changes the game engine to support the type of singleplayer/ multiplayer game the researcher requires. This parameter also changes the maximum and minimum number of actors can be present in a game at the same time. Following are the list of competition types supported by our system.

  1. One Vs One - Two players competing against each other
  2. Group Vs Group - Two groups competing against each other. Both the group has same number of members.
  3. Group Vs Computer - A group competing against the current best score.
  4. Singleplayer - A single player competing against the current best score
  5. Multiplayer - A set of solo players competing against each other.

- **Number of Players**

  This configuration type defines the number of players who can play together at the same time. It is also dependent and constant for some competition type values. This parameter has two types.

1. Maximum number of Users
2. Minimum number of Users

Once the setup is done a researcher can then add the game to the list of online games in the website.

- **Game Selection**

  Whenever a researcher adds a game to the portal, an update is added to the live feed of the gamer as a notification. A Gamer can also check the list of live games at any time. All the configuration details set by a researcher will be visible to the gamer. A gamer can choose a game after checking the game configurations and add it to his/her profile. Then a gamer can start the MMO Game arcade application

- **MMO Game Arcade Application**

  This is a desktop based real time game application. The application is used to play the games. The application has a robust multiplayer game networking framework installed. Using this application. A player can host a game via Local area network or join a game available on the local network. The application also has support to massively multiplayer game online where a player can host a game online, and other players can join the game via Wide area network. A separate server is used as a middleman who has all the network related information about the players such as LAN IP, WAN IP, MAC address, etc. This server does not support the matchmaking of LAN IP to WAN IP to support the synchronisation of the multiplayer game. When the game is being played, the user play data related to all the previously chosen data types by the researcher are collected. Once the game is over the collected data is send to the server, and the server makes it available for the Researcher to view.

- **Game Play Data Reports**

  When a researcher adds a game to the web portal or keeps the game status "Live". The data about all the previously chosen data types, are continuously tracked for every time game is played and stored on the server and the viewing authority is only given to the researcher. A researcher can at any time view all the data as a report on the web portal. The researcher can also download the reports in any of the three supported formats, XML, JSON, Text.

Chapter 4

Architectural Planning and Crowd Evacuation

4.1 Encoding architectural standards

The intention of gamifying architectural design is to make the game usable by a player with any level of architectural knowledge. To meet these ends, the game framework provides a means of specifying and designing constraints/rules in the layout design phase of our game such that even casual player can achieve a layout which follows architectural standards.

Any architectural layout primarily comprises of walls, pillars, and doors. Hence we provide the tools to the player to modify and create these elements in the layout. However, the game also provides the tools to create gamified levels with different forms of challenges. An architect or designer may choose to focus on an entire design and formulate levels in terms constraints on available elements and modifications in increasing levels of difficulty. The architect or designer may formulate levels in terms of separate portions of a larger environment and sort them in increasing difficulty, e.g. from the bathroom area, to a common area, to a high traffic area, and so on. The architect or designer may also choose to iteratively incorporate more portions of the environment such that success criteria in the new level may be additive in terms of the previous design. The gamification framework and level specification afford a number of options in order to lead the player to produce designs for the architect or designer. Display messages can be provided to help casual players learn about architectural standards when their design violate these constraints. Thus, the game can be used not just to produce design but also as an educational tool for novice designers or casual players.

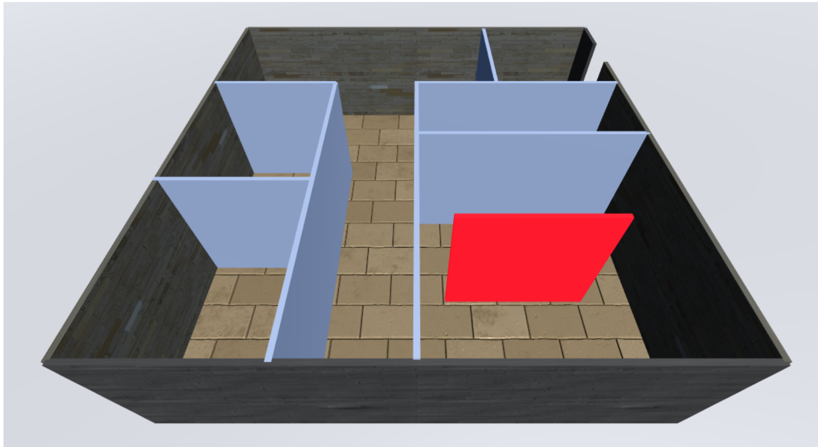


Figure 4.1: Environment design:invalid wall placement

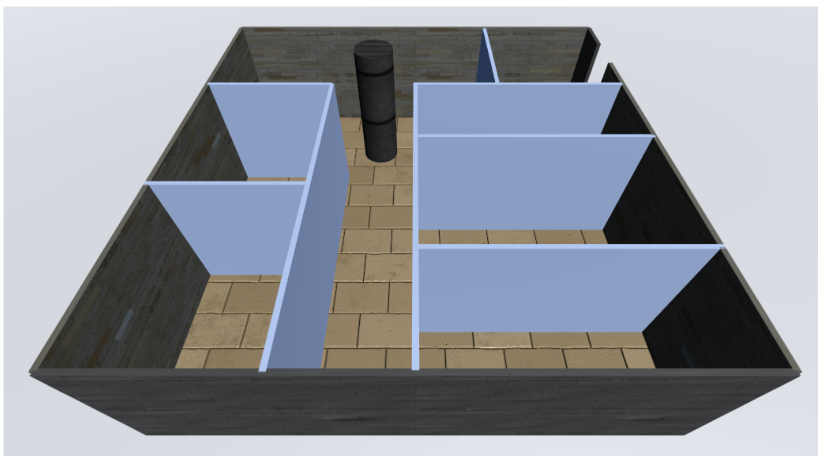


Figure 4.2: Environment design: wall placement

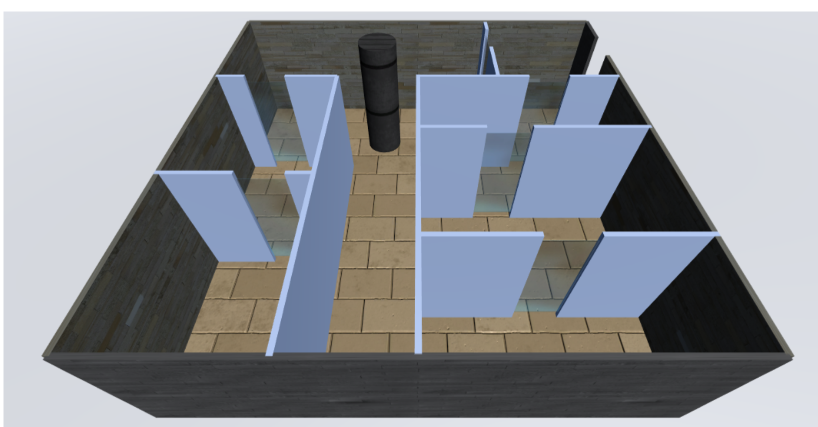


Figure 4.3: Environment design: pillar placement

4.2 Crowd Configuration and Simulation

To afford accurate crowd animations for player visualisation, a humanoid character model with animations is employed. The complete set includes idle and walking animations. The crowd members are generated and distributed at random locations in the layout based on the configurations set by the player. Destinations and distributions for locations of crowd members can be modified and constrained by the architect or designer as well. To ensure accurate navigation, the navigation mesh of the environment and path plans for each member are recomputed after every level change and before a simulation is started.

We used an agent-based model for simulating crowd steering. We have also tried to incorporate different steering decision behaviours observed in the crowd at the time of emergencies. For instance, during a fire, people tend to follow each other to find the exits or people tend to follow the exit signs more stringently than in their day to day lives. While going to their destination, if a member sees an exit which is closer than their assigned one their navigation goal switches to the closer observed goal. The members also change their destination and path when a big group of other members are moving towards a certain destination. So in a sense, the members follow the crowd unless they see an exit.

To afford rigorous and complete testing across dynamics and behaviours, the crowds are parametrized for aggression, homogeneity, and density. These parametrizations and a formal definition of a crowd are provided in the rest of this section.

- **Level of Aggression (LoA)** - Three degrees of aggression are provided; low, medium and high. The aggression degree determines the speed and acceleration of different members in the crowd. As the aggression level increases, the distribution of aggressive people in the crowd also increases. Fixed probabilities are associated with the low, medium, and high degrees and are used to assign appropriate speeds to the members. The mapping of degrees of aggression to speeds and accelerations can be seen in Table 4.2 and Table 4.3 respectively.

- **Level of Homogeneity (LoH)** - Three classes of homogeneity are available to the player: low, medium, and high. Changing this configuration changes the distribution of the radii of the crowd members. Low means that there is a little probability in the variety of members

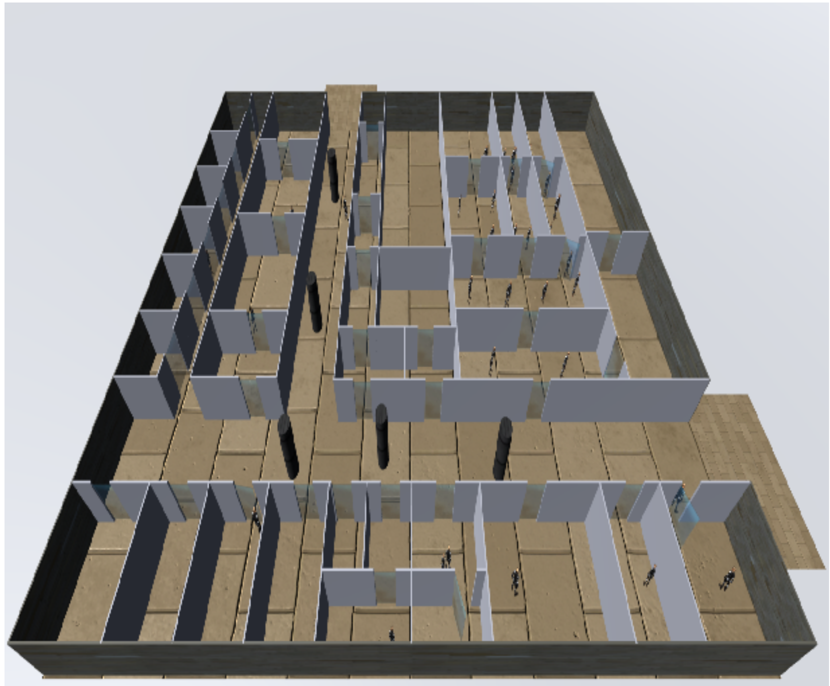


Figure 4.4: Crowd configuration and simulation: a) High Level of service

that have a radius other than the average one. This helps the player to identify how well their design handles different body shapes in a crowd.

- **Level of Service (LoS)** - There are six Levels of Service available to the player. Level of Service is used by traffic engineers to measure the quality of traffic flow both for automotive and pedestrian applications. LoS classes are typically given a grade level (from A-F), which are summarized in Table 4.1.

Examples of these parametrizations can be seen in Figure 4.4, 4.5, 4.6.

We formally denote a crowd $C = \langle g, k, l \rangle$ that describes the crowd related parameters where g, k, l denote LoS, LoA, LoH respectively. A gamified environment area is formally described as multiple heterogeneous sub areas. Let y_j be a random variable for sub area j with a value in range $[g_{min}, g_{max}]$. Let, n_j be the number of members in the sub area j.

$$\mathbf{n_j} = Q_j \times \delta \times x_i \tag{4.1}$$

where, Q_j is the area of sub area j and δ is the normalization constant. We described a transformation/mapping function that converts the crowd configuration to an instance of member

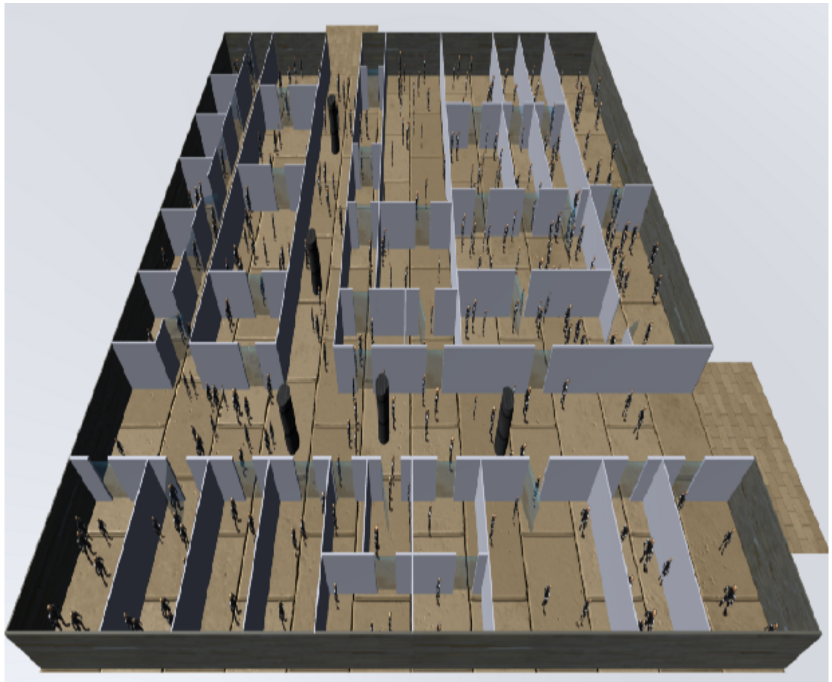


Figure 4.5: Crowd configuration and simulation: Low level of service

Figure 4.6: Crowd configuration and simulation: Level of Homogeneity

LOS	Crowd Density (people/square unit)
A	<=0.27
B	0.31 - 0.43
C	0.44 - 0.72
D	0.72 - 1.08
E	1.09 - 2.17
F	> 2.17

Table 4.1: LoS classifications and their associated densities.

k	x_i	v_a
Low	<=0.7	$S + SLOW \times x_i$
Low	>0.7	$S + SHIGH \times x_i$
Medium	<=0.5	$S + SLOW \times x_i$
Medium	>0.5	$S + SHIGH \times x_i$
High	<=0.7	$S + SHIGH \times x_i$
High	> 0.7	$S + SLOW \times x_i$

Table 4.2: Speed mapping for aggression levels. Let, x_i be the random variable for member i whose value is in the range [0,1]. SLOW and SHIGH are higher and lower aggression factors for speed respectively. S is the speed constant.

$A = \{a\}$ where each member $a = \langle \mathbf{p}, \mathbf{v}, \mathbf{f}, \mathbf{r}, \mathbf{h}, \mathbf{m} \rangle$ where p = position vector $(\hat{x_a}, \hat{y_a}, \hat{z_a})$, v = speed, r = radius, h = height, f = acceleration. $\hat{x_a}$ and $\hat{z_a}$ are random variables in range [x_{min_j},x_{max_j}] and [z_{min_j},z_{max_j}] respectively where x_{min_j},x_{max_j},z_{min_j},z_{max_j} are the four extreme coordinates of sub square area j.

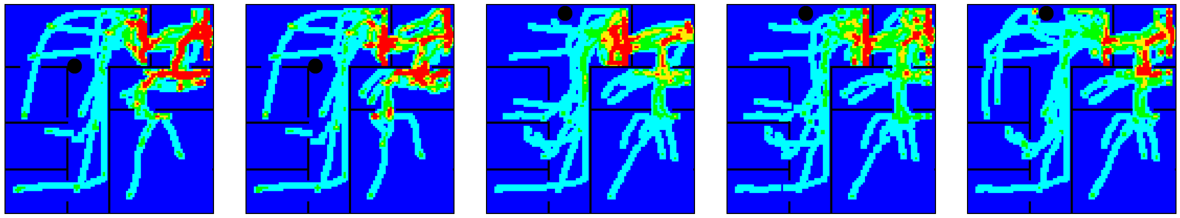
Formally, Level of Homogeneity is defined in terms of M,H,R as each crowd member's mass, height, and radius respectively. Let α, β, γ be heterogeneity factor for mass, height and radius respectively. The mapping of level to these factors can be found in Table 4.4.

k	x_i	f_a
Low	<=0.7	$F + ALOW \times x_i$
Low	>0.7	$F + AHIGH \times x_i$
Medium	<=0.5	$F + ALOW \times x_i$
Medium	>0.5	$F + AHIGH \times x_i$
High	<=0.7	$F + AHIGH \times x_i$
High	> 0.7	$F + ALOW \times x_i$

Table 4.3: Acceleration mapping for aggression levels. ALOW and AHIGH are higher and lower aggression factor for acceleration respectively. F is the acceleration constant.

l	x_i	m_a	h_a	r_a
Low	<=0.7	$M \times \alpha \times x_i$	$h \times \beta \times x_i$	$r \times \gamma \times x_i$
Medium	<=0.5	$M \times \alpha \times x_i$	$h \times \beta \times x_i$	$r \times \gamma \times x_i$
High	<=0.3	$M \times \alpha \times x_i$	$h \times \beta \times x_i$	$r \times \gamma \times x_i$

Table 4.4: Height, radius, and mass mapping for aggression levels.



(a) Time=16.615 (b) Time= 12.738 (c) Time=10.692 (d) Time=9.323 (e) Time=9.174

Figure 4.7: Iterative environment improvement, in terms of evacuation time, is shown from left to right.

4.3 Crowd-Aware Environment Analysis

A suite of analysis tools is made available to a player to visually validate their simulation results. In particular, we visualise metrics using heatmap so that the user can improve their design by identifying regions which are not performing well. The heat map is an orthographic view of the design containing all the elements(walls, doors, pillars) placed by the user and the path of all the agents. The level of crowd density of a region in the layout can be deduced by different coloured dots in the heat map. Figure 4.7 is an example of a heat map which shows the orthographic view of walls, doors, and pillars in the layout with aggregate crowds dynamics displayed as heat traces. If a region is not performing well, it will become congested and crowd density will rise and the region will become a deeper red. A player can improve the design to reduce the highlighted areas in the design.

4.4 Competitive Co-design of Environments

The game framework supports optimal environment design through an iterative and entertaining approach. A player can improve their design to reach optimality by iterating on subtle design modifications and running simulations. Furthermore, the framework supports collaboration and competitive co-design by exposing players to other's designs and performance analyses.

We have also provided the user with the ability to reinforce their design by learning from the best player's moves. One way that this is achieved is to show the best scorer's heatmap. A player can gain knowledge from that heatmap to potentially incrementally improve their design.

By allowing players to iteratively replay and view other’s analyses and designs the game fosters collaboration in our game which can guide players to a more optimal architectural design.

Chapter 5

Results

We hypothesise that with the gamification of a complex combinatorial problem in architectural design provides a user with an innovative means to improve their design and gradually reach optimality. We also hypothesise that competitive co-design is an effective solution to the non-convex optimisation problem that is crowd-aware environment design for efficient evacuation. We have conducted some user studies below to evaluate our hypothesis.

5.0.1 User Study

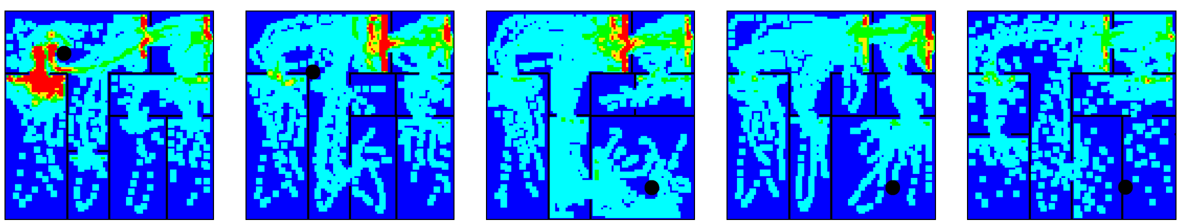
For our first hypothesis, we evaluated the evacuation times of particular user iterations for each of the 54 combinations of crowd configuration parameters. We have considered only the data of a user who has viewed the heat map for every iteration. For the second hypothesis, for each of the game level and each of the crowd parameter combination we have plotted all the evacuation times of user simulation runs in order of time of execution. We have considered the data of the users who have viewed the best scorer's heat map.

Each player is identified by a unique id assigned to him/her. Some of the user data we gathered are Player id, User placed architectural element positions, Total evacuation time of the simulation, User choices of crowd configuration parameters (LoS, LoA, LoH), User's choice of viewing the heat map, User's choice of viewing the Top Scorer's heat map.

Procedure

We have provided the user with an existing layout at the start of the level which helps us increase the difficulty as the user gradually reaches higher levels. Users cannot manipulate these already placed elements. Users can create a wall by clicking on two points in the layout. Pillars can be created by clicking at a point, on the floor, where the player want to place the pillar. The radius of the pillar can be changed by playing around with the slider in the creation menu. Doors can be created by clicking on the walls where the user wants to place the door. Walls, doors and pillars created by the user can be deleted by selecting them again. There are keyboard keys assigned to increase or decrease the width of the wall to be placed. Walls can be moved around by clicking them and using the arrows keys or by dragging them with the use of a mouse.

There are several constraints associated with each element in the layout. First of all the users cannot modify the elements of the layout existing when the level started. Users can only manipulate the elements they added to the existing layout. There are a fixed number of walls, pillars and doors that can be placed by the user. Walls can be placed only if it forms an enclosed space meaning that a wall must be touching walls on either end. A wall cannot be made to pass through the outside wall. In case the wall is placed incorrectly by the player, the wall will be marked red, and the player will not be able to start the simulation until the errors are rectified. If the width of the wall is less than two units, the wall will not be created.



Time=16.4175 Time=13.695 Time=11.1705 Time=9.669 Time=8.151

Figure 5.1: Current best player's heatmap with evacuation time for the crowd configuration LoS=F,LoA=High,LoH=Low in level 1

5.0.2 Results

In order to evaluate our hypotheses, we hosted our game on our website and asked people to play the game. We collected data for different users and different levels. Analysing and comparing the results, we observed the following.

After analysing the evacuation times of each user iterations for each level and each combination of crowd configurations, we have found a gradually decreasing curve of evacuation time in 73.33% of total 60 user cases. In Figure 4.7 we demonstrate a single case of iterative

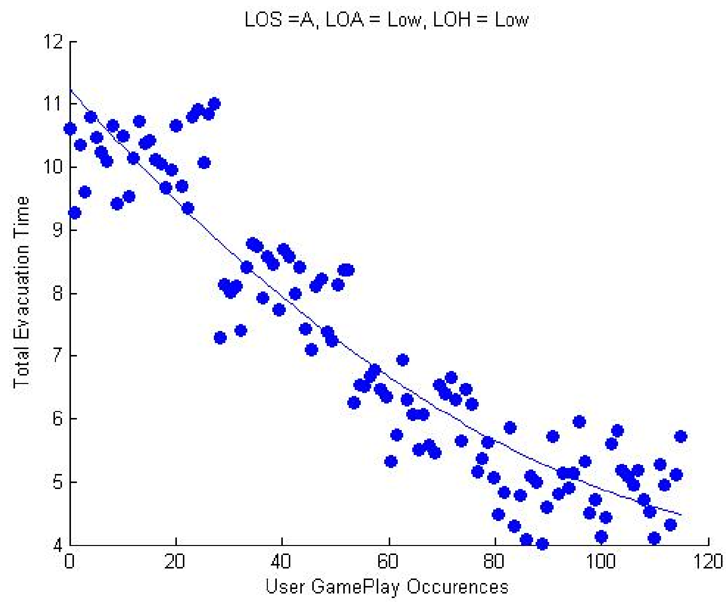


Figure 5.2: Evacuation Times for crowd configuration of LOS=A,LOA=Low,LOH=Low

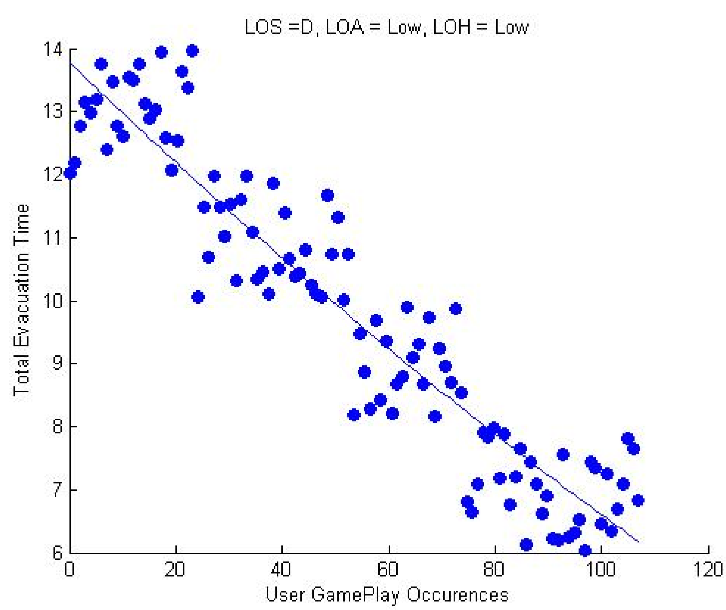


Figure 5.3: Evacuation Times for crowd configuration of LOS=D,LOA=Low,LOH=Low

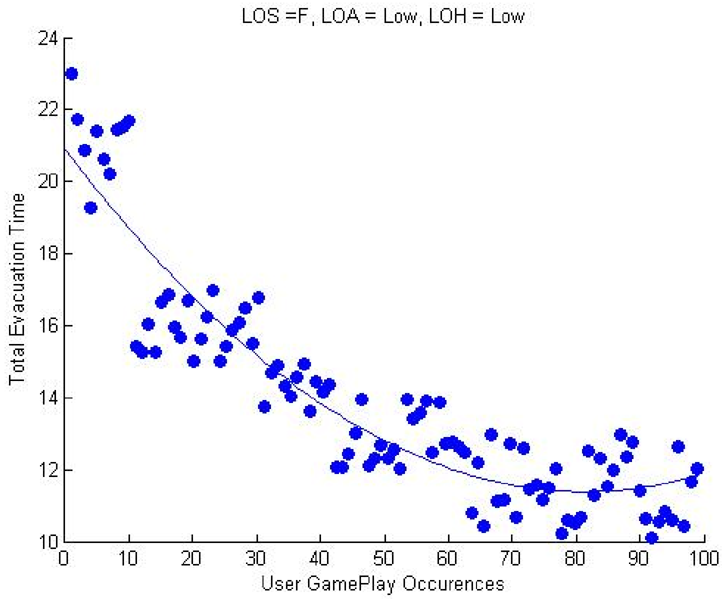


Figure 5.4: Evacuation Times for crowd configuration of LOS=F,LOA=Low,LOH=Low

improvement of a design by a player using the crowd configuration Los=crowd density of 1.09-2.17, LoA=High, LoH=Low.To prove that we used the following method.

For each user, we have calculated the following and plotted.

$$\sum_{i=1}^{n_A-1} I(i+1,i) \tag{5.1}$$

where n_A is the total number of iterations by a user A and

$$I(i+1,i) = \begin{cases} 1 & E_{i+1} \text{ - } E_i < 0.1 \\ -1 & E_{i+1}\text{-}E_i \geq 0.1 \end{cases}$$

where, E_i is the evacuation time of user A's i^{th} iteration. The threshold of difference in evacuation time between two consecutive iterations for a positive case is 0.1. We have then plotted the summation values from equation (2) for each user shown in histogram 5.5. The bars beyond the 0, represents number of positive cases and the rest are number negative cases.

For the second hypothesis, it was found that the evacuation times of different players viewing the current best scorer's heat map, gradually decreased and converged towards optimality. The curve formed by these players is very similar to that of a fitness function in computational optimisation strategies. Figure 5.1 is an example of this curve in which several players' iteratively

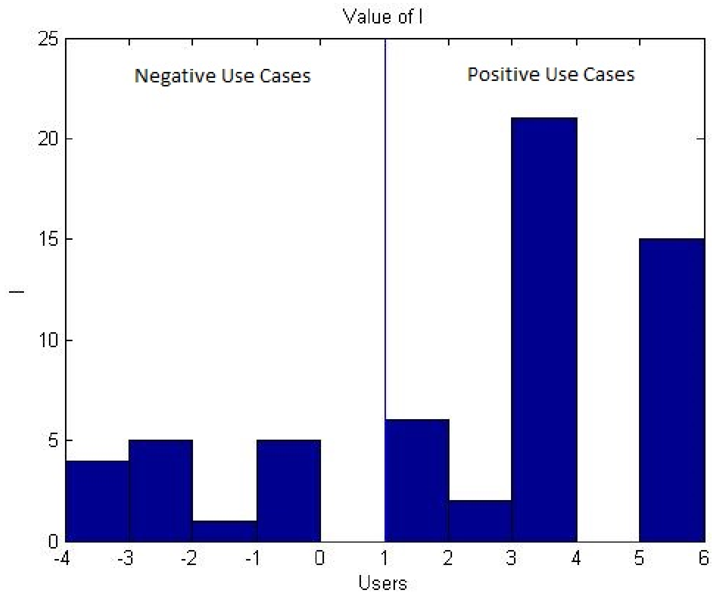


Figure 5.5: The bars represents number of user. The positive user cases are beyond 0

improved design by using the current best player's analyses and environment layout information. The figures 5.2, 5.3, 5.4 show the Graphs of Evacuation Times for different User Gameplays in order of occurrence for three separate combinations of crowd configuration parameters.

5.0.3 System Usability Results

We are using the System Usability Scale to calculate the System Usability Scale(SUS) score [12]. We received feedback from 61 players and calculated the score depending upon their responses. Our average usability score is 78.69 out of a scale of 100. Hence, our usability is average, but there is a lot of scope for improvement.

Chapter 6

Massively Multiplayer Online Game System

In chapter3 section 3.2 the general framework of MMO(Massively Multiplayer Online) game system is introduced. In the chapter, we explain how we have achieved the proposed advantages and usefulness of our system.

6.1 Connecting Researcher and Gamers

One of the most important aspects of our system is the connective virtual network between Gamers and Researchers. As part of our system, we have implemented a web portal for the users. A proper authorization has been provided to the website for data safety and security. The website can have two types of users: Gamers and surveyor. The services and presentation of the website are dynamically different based on the type of user.

In order to make the website dynamic, active and updated, we have introduced live feed to each user as shown in figure 6.1a. By default, everyone gets an update in their live feed if a Surveyor creates game. Besides this, an user can follow another user as shown in figure 6.1b. The following user will then get updates about the followed user. To provide more connectivity to the environment, we have provided a people search capability to every user. So, that a user can find other users on the website as shown in figure 6.2.

To keep a gamer updated about a newly created game by a surveyor, we have provided multiple capabilities on the website. A gamer can view all the live games in their portal. The gamer can follow the link to the games and add to his/her profile. A gamer can also view all the games added to his/her profile. On the other hand, a Surveyor can also view all the created games by him/her to find the game to modify or remove.

a) b)

Figure 6.1: a) Live Feed, b) Follow an user

Figure 6.2: People Search

6.2 Engaging gamers with the element of competition

The second biggest challenge to making our system successful in meeting the proposed requirement was to attract the gamers to play the games and keep then engaged to it which will help the researchers to accumulate enough data from their user study, to sum up with a provable hypothesis.

To keep the users engaged the primary need is to make the gaming experience simple and smooth. So that any gamer can play the game and easily adapt to the controls. A researcher might require data from users with all type of gaming skills, but the aim with higher priority is that the researcher should have a lot of user data. So we have kept the game tasks a simple as possible which require as much less controls as possible. The current games that are implemented and added to the framework are as follows.

- **Maze Runner**

  The game environment is a maze. At the start of the game, all the users are spawned randomly outside the maze. There are a set of objects that are randomly spawned at different locations inside the maze. The task of the game is to find the objects inside the maze and collect the object. We have used the Unity3D collision detection module to keep track of collected objects

- **Crowd Evacuation Race**

  The game environment is a maze. At the start of the game, all the users are spawned at the centre of the maze. The task of the game is to find a way to exit the maze.We have used the Unity3D collision detection module and assigned then to invisible objects barricading the exit to detect players exit from the maze.

- **Bomb Squad - Drones**

  The game environment is a city. There is a swarm of drones and a group of soldiers. The soldiers represent the bomb defusing squad, and the drones are used to detect the area of a planted bomb. A drone can be able to detect a bomb when in the range of its radiation, and it assumes and marks the area where a bomb is planted. It won't be able to find the exact location. Once the area is detected a user can inform the bomb squad with the coordinates of the detected location. Once the bomb squad reaches the location, a player can choose a soldier to control directly and search around the marked area for the bomb and once found, defuse it. So the main task of the game is to defuse bombs.

  A single or multiple drones can be selected, and a player can notify the place to go by clicking on the destination location. We have implemented the pathfinding ability for the drones using Unity3D Navigation mesh modules. A soldier or a group of soldiers can be selected as similar to the drones, and they can be moved to a destination as like as the drones. We have also implemented third person controller movement and assigned to the soldiers, but those will be activated only when the soldier is within the detected region around a bomb. To implement the bomb detection by drones, We have used the Unity3D collision detection module and assigned to an invisible large spherical object around the bomb representing the wavelength of the radiation of the bomb. If a drone collides with this hidden object, the area is marked.

- **Bomb Squad - Helicopter**

  This game is another version of the game "Bomb Squad". The game environment is also a city. Instead of a swarm of drones, a player is provided with a helicopter and a group of soldiers. The soldiers represent the bomb defusing squad, and the helicopter is used to detect the area of a planted bomb. A helicopter can be able to detect a bomb when in the range of its radiation and it assumes and marks the area where a bomb is planted. It won't is able to find the exact location. Once the area is detected a user can inform the bomb squad with the coordinates of the detected location and call the bomb squad. Once the bomb squad reaches the position, a player can choose a soldier to control directly and search around the marked area for the bomb and once found, defuse it. So the main task of the game is to defuse bombs.

  We have implemented the third person controller for the helicopter a player can notify the place to go by clicking on the destination location. We have applied the pathfinding ability for the drones using Unity3D Navigation mesh modules. We have applied the pathfinding and locomotion capability using Unity3D Navigation mesh module. We have also implemented third person controller movement and assigned to the soldiers, but those will be activated only when the soldier is within the detected region around a bomb. To implement the bomb detection by drones, We have used the Unity3D collision detection module and assigned to an invisible large spherical object around the bomb representing the wavelength of the radiation of the bomb. If a drone collides with this hidden object, the area is marked.

6.3 Polymorphic nature of a game based on parameter change

As we have expressed earlier that sometimes a researcher finds a game quite close to the research requirement and can be perfectly suitable for the study if small changes to the game configuration are made. So we have provided the capability to change the game according to the value of a set of parameters.

To achieve this, first of all, we have gone through many real-world games and tried to find the most common features in every game. Any small changes to these features can easily change the game to create another game. In order to accommodate this polymorphic ability to the games based on different parameter values, we have kept the core game tasks a generic and straightforward as possible. After careful research, we have come across the following set of configurations which are most common in every game.

6.3.1 Game Ending Strategy and Winner criteria

Game Ending Strategy defines the condition to be met in order to let the game end. The winning criteria are also directly Dependant on this configuration value and vice versa. The Gamed ending strategy of one game can be used as winning criteria in another game. In such case, the other strategy will take the opposite role. So the Game Ending Strategy and Winner criteria have an interchangeable role in games. After studying a lot of games, we have found out that the most common values for a game ending strategy are as follows.

- **Time Completion**

  In the games with time completion strategy, at the start, a limited amount of time is provided to perform some tasks, and the game ends when the assigned time is over. The Game ending condition is when Time is zero. In these games, the winning criteria are based on the amount of task completed in the limited time. The above mentioned four games included in our system, changes as follows with the variation of this configuration.

  - **Maze Runner**

    Game Ending Strategy is time completion and Winner criteria is number of objects collected.

  - **Crowd Evacuation Game**

    Game Ending Strategy is time completion and Winner criteria is number of times the user is able to exit the maze within that time. In our implementation we make the user go back to the mid position of the maze once they reaches the exit.

- **Bomb Squad - Drones**

  Game Ending Strategy is time completion and Winner criteria is number of bombs defused.

- **Bomb Squad - Helicopter**

  Game Ending Strategy is time completion and Winner criteria is number of bombs defused.

- **Task Completion**

  In the games with task completion strategy, there is a timer assigned to the game which keeps on counting the time from the start of the games. The player is assigned some tasks to perform. When all the assigned task are completed the game end. So the game end depends on the completion of the tasks assigned. In these games the winning criteria is based on amount of time taken to complete all the assigned tasks. The above mentioned four games included in our system, changes as follows with the change of this configuration.

  - **Maze Runner**

    Game Ends when player is able to collect all the objects and Winner criteria is time taken to collect all the objects.

  - **Crowd Evacuation Game**

    Game Ends when the player reaches the exit of the maze and Winner criteria is the time taken by the player to exit the maze.

  - **Bomb Squad - Drones**

    Game Ends when player have defused all the bombs and Winner criteria is the time taken to defuse all the bombs.

  - **Bomb Squad - Helicopter**

    Game Ends when player have defused all the bombs and Winner criteria is the time taken to defuse all the bombs.

6.3.2 Competition Type

This configuration is critical to modifying a game based on the research requirement specifically. This criterion also allows us to the ability to attract the users by the inclusion of the element of competition. Type a game dynamically changes which the value of choice for this configuration parameter. In implementation perspective, this configuration is also related to the other settings. After careful research on real world games, we have found out the following competition types.

- **One on One**

  If the competition type is one on one then game will have only two players and both will act as an opponent to the other. The winner is the player with better score. The better one is chosen based on the game ending strategy or winning criteria.

- **Group Vs Group**

  If the competition type is one on one, then the game will have only two players, and both will act as an opponent to the other. The winner is the player with a better score. The better one is chosen based on the game ending strategy or winning criteria.

- **Group Vs Computer**

  If the competition type is Group Vs Computer, then it is similar to a singleplayer where a player competes with the current best score and tries to beat that score. Here, in this case, a group of players together attempts to beat the current best score by a team. The winner is decided only when a team surpasses the current best score. The better team is chosen based on the game ending strategy or winning criteria.

- **Single Player**

  If the competition type is Singleplayer, then a player competes with the current best score and tries to beat that score. The single game there will be two teams playing against each other. The winner is decided only when the player surpasses the current best score. The better team is chosen based on the game ending strategy or winning criteria.

- **Multiplayer**

  If the competition type is Multiplayer, then a set of players compete. Each player participates in the game as a solo. The winner is the player with the best score. The better team is chosen based on the game ending strategy or winning criteria. The scores of the participants are displayed at the end of the game in order.

6.3.3 Number of players

The configuration parameters to set or change the number of players are vital in the case of a multiplayer game. There are two types of configurations representing the number of players.

- Maximum number of players
- Minimum number of players

In the case of a multiplayer game, the minimum number of players value determines how many minimum numbers of players has to join the network lobby to start the game. The maximum number of players value represents how many numbers of players maximum can participate in a game. This configuration can be very helpful for a researcher to manipulate the game based on the number the actors needed to perform a survey. The number of players is constant in the case of competition type one on one (both values set to 2) and Single player (both values set to 1). In the case of group vs. group, the number of players represents the number of players in a group. The total number of players in the game in such case will be double of the number of players.

6.3.4 Difficulty

Game difficulty is one of the most popular and default configuration in a game. The difficulty value of a game will determine the perfection of the game AI or increase/decrease in obstacles or increase/decrease in the amount of task etc. Choice of this value will manipulate the game within the range of predefined changes.

6.3.5 Game Environment

The game environment is also an important aspect.Changing which we can manipulate a game to meet a research requirement. A set of predefined Game environments is provided in our system. A single game environment can be chosen for a game. The game set of choices for a game environment also changes based on the type of game selected.

In the researcher web page in our web portal, a researcher can create a game by selecting the values for the configurations mentioned above. A gamer would not be able to change the configuration. The gamer will only be able to view all the information about the values of the game configuration set by the researcher. When a researcher creates a game, a gamer will receive a notification about the new game. Once the game is live, any gamer can play the game. A researcher cannot delete a created game if it's added to any gamer profile, but he/she can take it down from live.

6.4 Robust Massive Multiplayer Online Environment

Our system is mainly built around a robust MMO(Massive Multiplayer Online) environment. As our basic aim is to provide a smooth and consistent MMO gaming experience, we have taken extra care in the robustness of the MMO environment. We have used and extended the Unity3D Networking framework UNET to create our MMO environment. The implementation of the MMO framework consists of two parts as follows.

6.4.1 Network Lobby

Before entering into the network environment, a set of prerequisite values to be set up which are crucial for an MMO environment setup. Our system provides three types of setup for an MMO environment. The Network lobby contains sections for each of these three setups. The three kind of MMO setups is as follows.

- **Play and Host Online**

  This setup is mainly to connect players together in a single multiplayer game using wide area network. By choosing this option, a user can decide to be the host online as well as a player. In this case, the other users who want to join this game can do so without knowing the IP address of the Host. In this setup, a server situated somewhere else acts as a middleman between the users. This server the network matchmaking i.e. this server takes care of the matching wan IP address with LAN IP address and intercommunication between the players to synchronise and maintain the MMO game state. We have used Unity3D multiplayer services to use it as our network match making server.

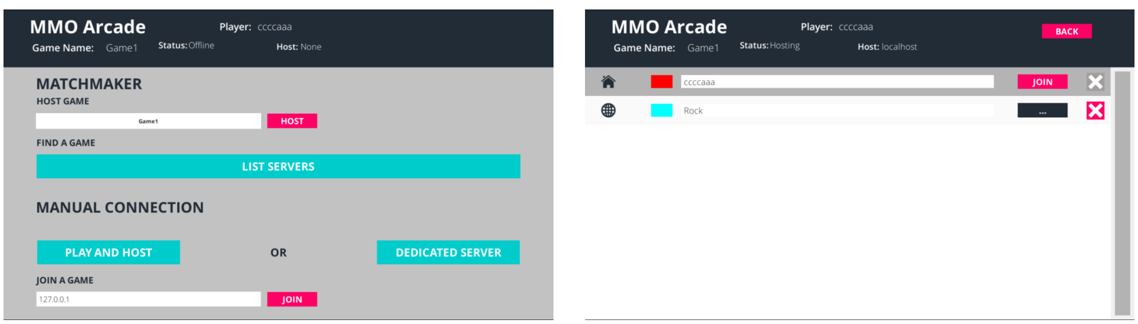
- **Play and Host via LAN**

  This setup is mainly to connect players situated in a Local area network. By choosing this option, a user can choose to be a host as well as a player. In this case, the other users who want to join this game need to know the LAN IP address of the host.

- **Dedicated server via LAN**

  By this setup, a user can only act as a dedicated server in his/her Local area network. Other players can connect to the game using this host's local Ip address.

Once any other the setup mentioned above is chosen the game lobby s opened and users start to join in. When a new user joins the lobby, all other users are able to see that. All the users have to wait in the lobby until the minimum required number of user joins the lobby. Once that many numbers of user joins the lobby, players can start the game.The network lobby is shown in figure 6.3.



a) b)

Figure 6.3: a) Main Lobby , b) Game Lobby

6.4.2 Network Environment

The network environment is combined with the game environment. We have used and extended UNET framework to setup the network environment. In the network environment, we have setup implementations to synchronise the player and non-player object positions, rotations, parameters value about animator state and other game parameters representing the game state. We have maintained a time gap between sending of update messages and this time gap varies based on the priority and impact of the parameters whose values are to be sent. The player synchronisation in our multiplayer environment is shown in figure 6.4 and 6.5.

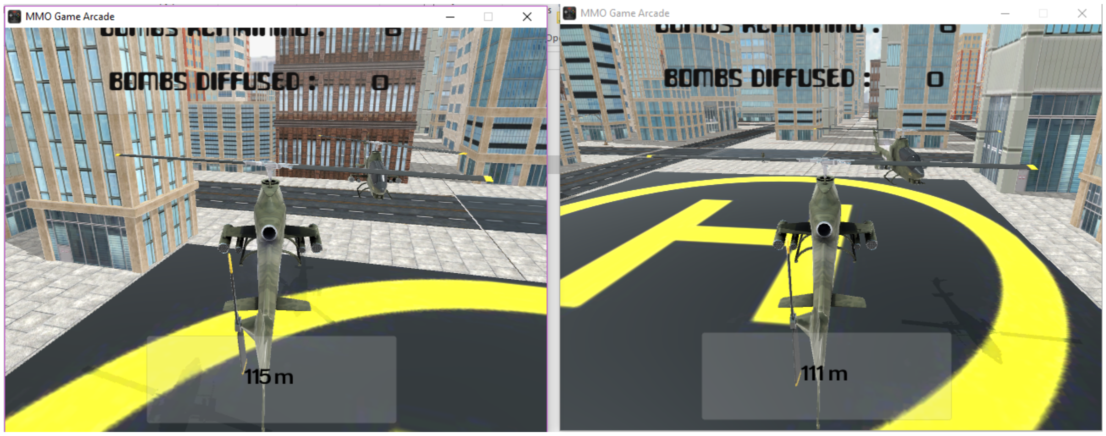


Figure 6.4: Multiplayer Position Synchronization

Figure 6.5: Multiplayer Animation Synchronization

6.5 Game play data Ownership and representation

Our system also provided the ability to choose the types of games data that are to be tracked during the gameplay. A researcher can select the different types of data to be tracked based on his research requirements. To maintain data authority, we have implemented the system in a way that a game data will only be available and visible to the creator of the game. So that there would not be any conflict of user study data between different researchers.

There are different types of choices for representation of data to be delivered to the owner.

A researcher can choose the format in which they want the data like a text file, XML file or as JSON etc.

Chapter 7

Conclusion

7.1 Contribution 1

Some of the shortcomings of our current crowd evacuation game were that some of the levels were quite tedious. Layouts in each level are designed manually. It would be much more efficient if this process were automated. All agents behaviour in the game is quite similar, and it would make sense to provide different behaviours to different agents. The game also needs more advanced architectural standards to be included as constraints to help casual users' designs to meet these standards. Future work includes making the simulations as realistic as possible with improved agent behaviour. Users can be provided with behaviour elements to add to their crowd agents along with the current crowd configuration parameters.

We think that our game can help the knowledge of architectural design go beyond the boundaries of architectural experts and can make the task of architectural design more interesting and entertaining. We also believe this paper is a beginning of more future architectural researches using serious games.

7.2 Contribution 2

Some of the shortcomings of our current MMO framework were that ability for the surveyor to design a game environment if provided with common elements. The current game control supports mouse-keyboard and joystick. In future, this framework can be combined with Virtual reality interfaces and motion capturing tools to observe and track actual movement of a player. A set of autonomous agents can be included as an obstacle to games in the framework which can act as a game difficulty factor.

We think that our framework can be a one stop medium for Researchers looking to test a real world event and perform a user study and Gamers who are looking an exciting and smooth massively multiplayer online gaming experience.

References

[1] AIZHU, T. F. R. Agent-based evacuation model incorporating fire scene and building geometry. *TSINGHUA SCIENCE AND TECHNOLOGY 21/25* (2008), 708–714.

[2] ALEXANDER SHOULSON, NATHAN MARSHAK, M. K. N. I. B. Adapt: the agent developmentand prototyping testbed. *IEEE Transactions on Visualization and Computer Graphics 20*, 7 (2014), 1035–1047.

[3] ARVIN, S. A., AND HOUSE, D. H. Modeling architectural design objectives in physically based space planning. *Automation in Construction 11*, 2 (2002), 213–225.

[4] BASSUET, A., RIFE, D., AND DELLATORRE, L. Computational and optimization design in geometric acoustics. *Building Acoustics 21*, 1 (2014), 75–86.

[5] BEGHINI, L. L., BEGHINI, A., KATZ, N., BAKER, W. F., AND PAULINO, G. H. Connecting architecture and engineering through structural topology optimization. *Engineering Structures 59* (2014), 716–726.

[6] BERGER, M. J., AND COLELLA, P. Local adaptive mesh refinement for shock hydrodynamics. *Journal of computational Physics 82*, 1 (1989), 64–84.

[7] BERSETH, G., HAWORTH, M. B., KAPADIA, M., AND FALOUTSOS, P. Characterizing and optimizing game level difficulty. In *Proceedings of the Seventh International Conference on Motion in Games* (2014), ACM, pp. 153–160.

[8] BERSETH, G., KAPADIA, M., HAWORTH, B., AND FALOUTSOS, P. SteerFit: Automated Parameter Fitting for Steering Algorithms. In *Eurographics/ ACM SIGGRAPH Symposium on Computer Animation* (2014), V. Koltun and E. Sifakis, Eds., The Eurographics Association.

[9] BERSETH, G., KAPADIA, M., HAWORTH, B., AND FALOUTSOS, P. SteerFit: Automated Parameter Fitting for Steering Algorithms. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2014), SCA '14, ACM.

[10] BERSETH, G., USMAN, M., HAWORTH, B., KAPADIA, M., AND FALOUTSOS, P. Environment optimization for crowd evacuation. *Computer Animation and Virtual Worlds 26*, 3–4 (2015), 377–386.

[11] BLOCK, P., KNIPPERS, J., MITRA, N. J., AND WANG, W. Advances in architectural geometry 2014, 2014.

[12] BROOKE, J. Sus: a quick and dirty usability scale. In *Usability Evaluation in Industry*, B. W. P.W.Jordan, B. Thomas and I. McClelland, Eds. London: Taylor and Francis, 1996, pp. 189–194.

[13] CHAMPANDARD, A. J. Modern pathfinding techniques. Tech. rep., AiGameDev.com, 2009.

[14] CURTIS, S., BEST, A., AND MANOCHA, D. Menge: A modular framework for simulating crowd movement, 2015.

[15] DIRK HELBING, I. F. . T. V. Simulating dynamical features of escape panic. *Nature 10.1038 407* (August 2000), 487–490.

[16] FAJEN, BRETT R.; WARREN, W. H. Behavioral dynamics of steering, obstable avoidance, and route selection. *Journal of Experimental Psychology: Human Perception and Performance, 29* (April 2003), 343–362.

[17] GLEN BERSETH, M BRANDON HAWORTH, M. K. P. F. Characterizing and optimizing game level difficulty. In *Proceedings of the Seventh International Conference on Motion in Games* (2014), ACM, pp. 153–160.

[18] GUY, S. J., CHHUGANI, J., CURTIS, S., DUBEY, P., LIN, M., AND MANOCHA, D. Pedestrians: a least-effort approach to crowd simulation. In *Proceedings of SCA* (2010), SCA '10, pp. 119–128.

[19] GUY, S. J., VAN DEN BERG, J., LIU, W., LAU, R., LIN, M. C., AND MANOCHA, D. A statistical similarity measure for aggregate crowd dynamics. *ACM Transactions on Graphics (TOG) 31*, 6 (2012), 190.

[20] HART, P. E., NILSSON, N. J., AND RAPHAEL, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics 4*, 2 (July 1968), 100–107.

[21] HAWORTH, B., USMAN, M., BERSETH, G., KAPADIA, M., AND FALOUTSOS, P. Evaluating and optimizing level of service for crowd evacuations. In *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games* (New York, NY, USA, 2015), MIG '15, ACM, pp. 91–96.

[22] HELBING, D., FARKAS, I., AND VICSEK, T. Simulating dynamical features of escape panic. *Nature*, 2000.

[23] HELBING, D., FARKAS, I., AND VICSEK, T. Simulating dynamical features of escape panic. *Nature 407*, 6803 (2000), 487–490.

[24] HELBING, D., AND MOLNÁR, P. Social force model for pedestrian dynamics. *PNAS 1016507108 108* (March 2011), 6884–6888.

[25] HSU, Y.-C., AND KRAWCZYK, R. J. New generation of computer aided design in space planning methods: A survey and a proposal. In *International Conference on Computer Aided Architectural Design Research in Asia* (2003), pp. 101–116.

[26] JIANG, L., LI, J., SHEN, C., YANG, S., AND HAN, Z. Obstacle optimization for panic flow - reducing the tangential momentum increases the escape speed. *PLoS ONE 9*, 12 (12 2014), 1–15.

[27] JIANG, L., LI, J., SHEN, C., YANG, S., AND HAN, Z. Obstacle optimization for panic flow-reducing the tangential momentum increases the escape speed. *PloS one 9*, 12 (2014), e115463.

[28] JOHANSSON, A., AND HELBING, D. Pedestrian flow optimization with a genetic algorithm based on boolean grids. In *Pedestrian and evacuation dynamics 2005*. Springer, 2007, pp. 267–272.

[29] JOÃČO EMÃ■LIO ALMEIDA, ROSALDO ROSSETI, A. L. C. Crowd simulation modeling applied to emergency and evacuation simulations using multi-agent systems. In *DSIEâĂŹ11 - 6th Doctoral Symposium on Informatics Engineering (pp. 93-104). Engineering Faculty of Porto University, Porto* (2011).

[30] KAI NINOMIYA, MUBBASIR KAPADIA, A. S. F. G. N. B. Planning approaches to constraintâĂŘaware navigation in dynamic environments. *Computer Animation and Virtual Worlds 26*, 2 (2015), 119–139.

[31] KAMKARIAN, P., AND HEXMOOR, H. Modeling crowd evacuation from indoor spaces. In *Proceedings on the International Conference on Artificial Intelligence (ICAI)* (2012), The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), p. 1.

[32] LERNER, A., CHRYSANTHOU, Y., SHAMIR, A., AND COHEN-OR, D. *Motion in Games: Second International Workshop, MIG 2009, Zeist, The Netherlands, November 21-24, 2009. Proceedings*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, ch. Data Driven Evaluation of Crowds, pp. 75–83.

[33] LIKHACHEV, M., GORDON, G. J., AND THRUN, S. Ara*: Anytime a* with provable bounds on sub-optimality. In *Advances in Neural Information Processing Systems* (2003), p. None.

[34] MARCELO KALLMANN, M. K. Navigation meshes and real-time dynamic planning for virtual worlds. *ACM SIGGRAPH 2014 Courses* (2014), 3.

[35] MATTHEW SCHUERMAN, SHAWN SINGH, M. K. P. F. Situation agents: agentâĂŘbased externalized steering logic. *Computer Animation and Virtual Worlds 21*, 3âĂŘ4 (2010), 267–276.

[36] MEHDI MOUSSAÃŘD, MUBBASIR KAPADIA, T. T. R. W. S. M. G. D. H. C. H. Crowd behaviour during high-stress evacuations in an immersive virtual environment. *Journal of The Royal Society Interface 13*, 122 (2009), 4–14.

[37] MEHDI MOUSSAÃŘDA, D. H., AND THERAULAZA, G. How simple rules determine pedestrian behavior and crowd disasters. *Phys. Rev. E 51* (May 1995), 4282–4286.

[38] MERRELL, P., SCHKUFZA, E., AND KOLTUN, V. Computer-generated residential building layouts. *ACM Trans. Graph. 29*, 6 (Dec. 2010), 181:1–181:12.

[39] MICHALEK, J., AND PAPALAMBROS, P. Interactive design optimization of architectural layouts. *Engineering Optimization 34*, 5 (2002), 485–501.

[40] MUBBASIR KAPADIA, SHAWN SINGH, B. A. G. R. P. F. Steerbug: an interactive framework for specifying and detecting steering behaviors. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2009), ACM, pp. 209–216.

[41] MUBBASIR KAPADIA, ALEJANDRO BEACCO, F. G. V. R. N. P. N. I. B. Multi-domain real-time planning in dynamic environments. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics symposium on computer animation* (2013), ACM, pp. 115–124.

[42] MUBBASIR KAPADIA, NURIA PELECHANO, J. A. N. B. Virtual crowds: Steps toward behavioral realism. *Synthesis Lectures on Visual Computing: Computer Graphics, Animation, Computational Photography, and Imaging 7*, 4 (2015), 1–270.

[43] MUBBASIR KAPADIA, N. I. B. Navigation and steering for autonomous virtual humans. *Wiley Interdisciplinary Reviews: Cognitive Science 4*, 3 (2013), 263–272.

[44] MUBBASIR KAPADIA, MATT WANG, S. S. G. R. P. F. Scenario space: characterizing coverage, quality, and failure of steering algorithms. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2011), ACM, pp. 53–62.

[45] MUBBASIR KAPADIA, SHAWN SINGH, W. H. P. F. Egocentric affordance fields in pedestrian steering. In *Proceedings of the 2009 symposium on Interactive 3D graphics and games* (2009), ACM, pp. 215–223.

[46] NIKOLAI W. F. BODE, ARMEL U. KEMLOH WAGOUM, E. A. C. Human responses to multiple sources of directional information in virtual crowd evacuations. *rsif.2013.0904 10* (November 2013).

[47] PELECHANO, N., ALLBECK, J. M., AND BADLER, N. I. *Virtual Crowds: Methods, Simulation, and Control.* Morgan & Claypool Publishers, 2008.

[48] PETTRÉ, J., ONDŘEJ, J., OLIVIER, A.-H., CRETUAL, A., AND DONIKIAN, S. Experiment-based modeling, simulation and validation of interactions between virtual walkers. In *ACM SIGGRAPH/EG SCA* (2009), pp. 189–198.

[49] POTTMANN, H., EIGENSATZ, M., VAXMAN, A., AND WALLNER, J. Architectural geometry. *Computers & Graphics* (2014).

[50] RIBEIRO, J., ALMEIDA, J. E., ROSSETTI, R. J., COELHO, A., AND COELHO, A. L. Towards a serious games evacuation simulator. *arXiv preprint arXiv:1303.3827* (2013).

[51] SHAWN SINGH, MUBBASIR KAPADIA, B. H. G. R. P. F. A modular framework for adaptive agent-based steering. *Symposium on Interactive 3D Graphics and Games*, 9 (2011).

[52] SHAWN SINGH, MUBBASIR KAPADIA, G. R. P. F. Footstep navigation for dynamic crowds. *Computer Animation and Virtual Worlds 22*, 2âĂŘ3 (2011), 151–158.

[53] SHAWN SINGH, MUBBASIR KAPADIA, P. F. G. R. An open framework for developing, evaluating, and sharing steering algorithms. *International Workshop on Motion in Games 5884*, 2 (2009), 158–169.

[54] SHAWN SINGH, MUBBASIR KAPADIA, P. F. G. R. Steerbench: a benchmark suite for evaluating steering behaviors. *Computer Animation and Virtual Worlds 20*, 5âĂŘ6 (2009), 533–548.

[55] SHI, X., AND YANG, W. Performance-driven architectural design and optimization technique from a perspective of architects. *Automation in Construction 32* (2013), 125–135.

[56] SINGH, S., KAPADIA, M., FALOUTSOS, P., AND REINMAN, G. An open framework for developing, evaluating, and sharing steering algorithms. In *Proceedings of MIG* (2009), Springer-Verlag, pp. 158–169.

[57] SINGH, S., KAPADIA, M., HEWLETT, B., REINMAN, G., AND FALOUTSOS, P. A modular framework for adaptive agent-based steering. In *ACM I3D* (2011), pp. 141–150.

[58] TREUILLE, A., COOPER, S., AND POPOVIĆ, Z. Continuum crowds. In *ACM Transactions on Graphics (TOG)* (2006), vol. 25, ACM, pp. 1160–1168.

[59] TURRIN, M., VON BUELOW, P., AND STOUFFS, R. Design explorations of performance driven geometry in architectural design using parametric modeling and genetic algorithms. *Advanced Engineering Informatics 25*, 4 (2011), 656–675.

[60] VAN DEN BERG, J., GUY, S. J., LIN, M., AND MANOCHA, D. Reciprocal n-body collision avoidance. In *Robotics Research*, vol. 70. 2011, pp. 3–19.

[61] WINTER, H. Modelling crowd dynamics during evacuation situations using simulation.

[62] WOLINSKI, D., GUY, S., OLIVIER, A.-H., LIN, M., MANOCHA, D., AND PETTRÉ, J. Parameter estimation and comparative evaluation of crowd simulations. *Computer Graphics Forum 33*, 2 (2014).

[63] YU, L.-F., YEUNG, S. K., TANG, C.-K., TERZOPOULOS, D., CHAN, T. F., AND OSHER, S. Make it home: automatic optimization of furniture arrangement. *ACM Transactions on Graphics 30*, 4 (2011), 86.

[64] ZAHARIA, M. H., LEON, F., PAL, C., PAGU, G., DEMIRALP, M., BAYKARA, N., AND MASTORAKIS, N. Agent-based simulation of crowd evacuation behavior. In *WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering* (2009), no. 11, World Scientific and Engineering Academy and Society.

[65] ZHENG, X., ZHONG, T., AND LIU, M. Modeling crowd evacuation of a building based on seven methodological approaches. *Building and Environment 44*, 3 (2009), 437–445.