University of Derby

School of Electronics, Computing & Mathematics

A project completed as part of the requirements for
BSc (Hons) Computer Games Programming

entitled

How Crowd Simulations Could Improve Our Security

by

Diego Marti Mason

May 2020

Abstract

The volume of individuals gathering keeps increasing exponentially. These can present a challenge to urban planners and architects when designing building layouts and evacuation plans, so that is why, researchers have been looking for decades solutions to phenomena that are likely to happen if panic spreads among a crowd.

The purpose of this study is to develop a crowd simulator that could improve crowd safety. We gather information from an extensive literature review and create a crowd simulator that aligns with the findings of past research. The methodology used presents different models of simulating crowds, using a multi-agent system, and a psychological based approach. And it is later validated with previous research by running simulations and analysing them. So, this study proves the efficiency of studying crowds and creating simulators that improves our security.

Acknowledgements

To my family and friends who are always there for me.

To my parents, Vero, and sister, who apart from their unconditional support, they have always encouraged me to follow my dreams and keep pushing forward. Without my parents sacrifices, and their fantastic job at raising me, I would not be studying here - I hope that, after having been a terrible student at school, I have made you proud.

To grandad, my role model, who I always look up to, there is not a day that goes by where I do not think of him, he was a hardworking and caring man, who went through many struggles, that many will not stand, but still was able to keep his head up and carry on with a smile. I am who I am because of him, my inspiration to be the best I can.

And to Dr. Chris Windmill, not just for supervising this project and contributing towards this study, but for his support during my time at University, his advice and knowledge has been vast and has helped me to become a better coder, thanks.

Table of Contents

1. Introduction

1.1. Project Rationale

It is very likely that in our day-to-day we will encounter big gatherings, we attend big events, we shop in massive shopping centres, we traverse big streets, and concentrate in diverse areas. As population keeps increasing, and human settlements grow bigger, it is likely for problems to appear. If correct planning has not been made, questions arise. What could happen when the unexpected occurs? How can we be assured that emergency routes work?

In an emergency crowds reacts in a multitude of ways and the response patterns vary with the situation, timing, and place. To many it may seem like they are acting irrational, yet this is not the case, as crowds react to the best of their knowledge [1]. Past events like the Hillsborough disaster in 1989, the worst sporting disaster in British history, show us the importance to plan when huge crowds are expected, safety routes and urban planning could have safe 96 lives [2].

Hence Crowd Simulators can help to those who work to keep us safe understand crowds' response to danger and develop better emergency routes that may reduce crowd-disasters in the future.

1.2. Project Aim and Objectives

This project aims to create a Crowd Simulation software within the Unity 3D Engine, chosen for its graphics and physics engine, that will help urban planners, emergency strategist, and others who may be interested in analysing crowd flows. To achieve so a large literature review is needed to understand the work that has been done so far, what needs to be considered about crowd behaviour and what improvements could be made compared to other current software. Additionally, to help those researchers and evacuation planners, the simulator will need to generate meaningful data, such as heatmaps and every entity's average escape times. Finally, the simulator needs to be tested and validated with previous studies.

2. Literature Review

2.1. Introduction

A lot of research in many fields had been carried out to analyse crowd's behaviours in crisis situations. Psychologists, architects, urban planners and emergency strategists are some of the many experts who benefit from studying the gathering of individuals. Great effort has been put into improving simulations that help to evaluate crowds and ultimately improve our security.

To understand how crowds are simulated a large amount of information has been drawn from social psychology, cognitive psychology, sociology and physiology research, in addition to direct analysis of different videos from real events. This information contains vital data to understand how the human brain and body reacts to imminent unexpected danger and how an individual interacts with others in these situations.

This literature review explores different algorithms that had been used in current crowd simulations software, and the different approaches such as flow-based, entity-based, or agent-based.

2.2. Psychology research

One of the first things that must be understood when studying crowd behaviour in crisis situations is panic. Psychologists like Anthony R. Mawson describes panic as:

"Inappropriate (or excessive) fear and/or flight and highly intense fear and/or flight"

Anthony R. Mawson, 2005 [1]

although it rectifies and specifies that a crowd may be seen as panicking by external observers but not by the crowd itself. To the members of the crowd it appears to be rational behaviour in a moment of danger - to move to a known exit in a seemingly disorganized manner so they (the individual) can escape the hazard. Information gathered for the literature review show that these behaviours can lead to herd behaviours. As Mawson states that for mass panic to occur two conditions have to be met: firstly "major physical danger is present or imminent"; secondly "escape routes are either limited or rapidly closing" [1]. However mass panic is not the only source of panic within the group - some members of the crowd may go into a panic state as they see other individuals fleeing, assuming there is an imminent danger nearby. This is defined as contagious behaviour [3] that it is spread through the whole gathering of individuals in a certain area where escape routes may be limited or/and those are swiftly closing.

If we analyse individual's behaviour during an evacuation or an emergency, we could argue that their decision-making at that time is logical based on their knowledge at the time of the event. A common example many psychologists use to back up this concept is fire [4], they disagree with the idea that an individual would act irrationally in such an emergency, instead their decision will be based on their knowledge and understanding about fire and the location they are in. Although those decisions do not necessarily make the induvial right and could potentially put them in more danger, thus external observers judge them as irrational. Doctor G. Proulx states in his research on Occupant Behaviour and Evacuation [4] that is unlikely that individuals within a crowd would choose to stampede to safety or to crush each other, however, the crowd as whole may do so. Even so, there are many factors that can affect the behaviour of a group of humans in those kinds of situations, their personality will make them assume roles during the course of the event [4], some of them will take the role of leader, others will follow a leader, some others will copy others' behaviour.

Nonetheless their logical behaviour will change as the incident unfolds as individuals become aware of events e.g. [4]:

- when they start perceiving smoke, crowd will worry
- when the fire is seen by individuals:
    - in that moment they are aware of the magnitude of the event
    - nervousness to get out of the situations starts to appear
    - individuals will start rushing to an exit.

This can lead to a phenomenon known as "freezing by heating" [5] due to the blockage caused by fluctuation of an escape route and another phenomenon called "faster is slower" [5] result of the individuals' impatience causing clogging and reducing the overall speed of the crowd.

From the literature review the development of an individuals' behaviour in other crisis situations would be similar to a fire incident. Data gathered indicates that if a sound or any other sense is not perceived it will take time for individuals to trigger their "evacuation sense". This effect can clearly be seen when an alarm goes off, the common individual reaction is to wait either for the alarm to stop or wait until somebody in the crowd reacts [4]. This behaviour contradicts the purpose of an alarm signal as individuals should immediately react to those, Doctor Proulx believes that this is due to the lack of commitment towards it, and alarms systems relying on an assumption of human reactions [4]. Is not until a member of the crowds reacts or someone tells the crowd that they must evacuate when actions are taken [3], so it could be certainly said that there is a delay between the start of the incident and the crowd's reaction time.

The characteristics of each individual will play a huge part in a crisis. Such as gender, age, mobility, its current knowledge about the area or fires. These play a huge role when it comes to evacuate, certain circumstances like not knowing the area could lead the individual to leave by following the same path that they used for entering, even if they are closer to another exit. Being with known individuals (like family) will make individuals to stick together even if doing so possesses a threat to their own security. Contrary to common belief members of a crowd will tend to be altruistic in an emergency and rarely they will not attend those in distress [1].

2.3. Herding and other phenomena

Herding [6] is a common evacuation behaviour; it is a phenomenon that it is likely to happen when crowds are in a state of panic and all individuals are moving to the same exit. It could be defined as blocking an exit due to individuals acting irrational by blindly following other behaviours and its same path, this is caused by the nervousness.

We can observe in the figure below *(figure 1)* that two available exits are within reachable distance, but one of them gets flooded due to the nature of herd behaviours, thus the exits are not efficiently used, sooner or later this will increase the overall evacuation time and could jeopardize survival of others.

Professor Helbing suggests what would happen if an exit is jammed and opposite to it another exit is within range. Eventually some individuals will detect those exits and try to escape through them, causing a movement in the opposite direction hence obstructing more the path [7].

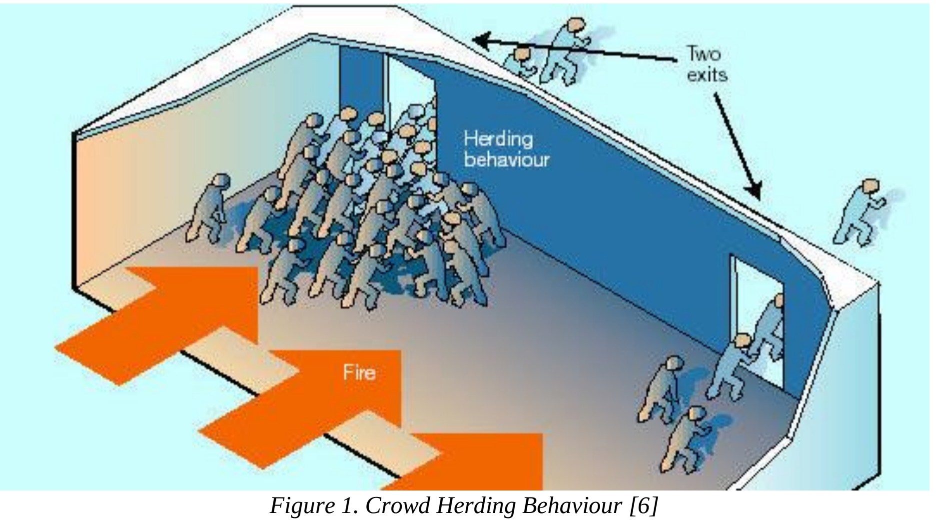

Figure 1. Crowd Herding Behaviour [6]

Another phenomenon observed by Helbing is when individuals start pushing each other and accelerating towards an exit a bottleneck occurs, due to these interactions a pressure of up to 4,450 N m^{-1} builds up [7]. This pressure can start bending metal or pushing bricks, moreover new obstacles will be created as individuals fall or get injured [7]. All this added to a herd behaviour will create an **arch-shape** at the exits as observed in figure 2, where each dot represents an individual. A semi-circular arch is formed causing clogging at the door and the evacuation times being increased [7].

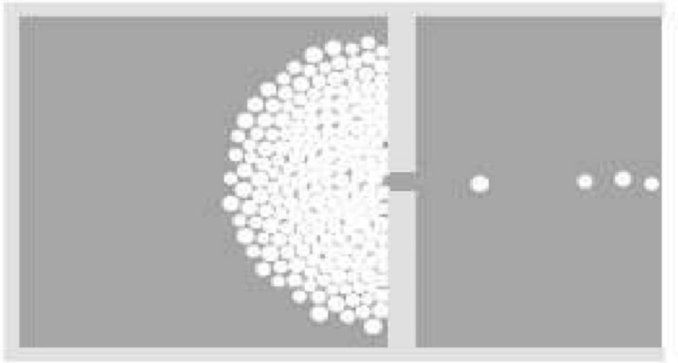

Figure 2. Arch-Shape Clogging [6]

Clogging [6][7] occurs when multiple individuals tussle to pass through a door or corridor slowing down their time to escape as well as individuals behind them, this cycle will repeat over and over increasing the shape of the arch.

In evacuations normally at **narrow corridors, bottlenecks** [7] will be formed as they get flooded by individuals, this will yet again reduce the overall speed of the crowd and create an arching effect as described above, yet another effect is observed by researchers from the University of North Carolina known as **wake effect** [8], this occurs after the individuals have gone through the narrow corridor, they will spread creating pockets of empty space and will not take advantage of all the space available. This could later unleash into more jamming.

2.4. Previous Work

So far, I have talked about the psychology research done over the past years, but, there has been a lot of work put into the computation area in addition to many algorithms to look at.

Unfortunately, we cannot rely in drills, people do not take them seriously and getting data out of them have been proven ineffective, as they lack from a real danger and crowds will not go into a panic state thus not giving realistic results [6]. Therefore the important of computer scientist to simulate crisis situations in the most realistic way possible, these algorithms gained a great importance in the late 90s and 00s, the reasons are various, one of them being the fast-paced hardware technology changes, therefore becoming easier to simulate certain number of “intelligent” agents and others of due to architectural and emergency leading to an increase of crowd disasters as the Hillsborough disaster [2].

Murakami et al. [9] proposes a model based in human interaction, they pointed out that so far algorithms back then had not considered the role of leadership, consequently proposing two different models to tackle this problem called the “Follow-direction method”[9], where a leader would give instructions to different individuals and then proceeded to an exit, and the “Follow-me-method” [9] where a leader would indicate those near him to follow him towards the exit, without letting them know where the exit was. They simulated these two models with sixteen individuals, the results suggested that the “follow-me-method” was ineffective as eleven evacuees had been taken to the wrong exit, thus putting them in more danger. More tests were carried out with more than one leader, resulting in confusion between individuals in the “follow-direction” method as some of them received multiple instructions that contradicted each other, the “follow-me-method” prove to be more efficient, still there were some individuals who evacuated through the wrong exit.

Although, I can see the reasoning between these two models, yet they miss a key factor which is that individuals will follow a group of others, therefore potentially proving that both methods could be efficient in all circumstances, unless wrong instructions or directions are given.

Other papers that I survey from late 1990s and 2000s indicates to me the hardware limitations of those decades, where only few agents could be simulated without putting a huge workload on the CPU.

There is multiple way to model crowd dynamic, some the most widely use are [6][10][11][12][13]:

- **Flow-based:** This model consists in a network of nodes, these nodes can be either physicals, destinations, or humans. These are based on particles; all its parameters and destination are pre-computed before the simulation runs based on the minimum time to evacuate. There is not change during the simulation, meaning that crowd behaviours described in past sections will not be considered, in addition the environment is fixed so it cannot be changed during the simulation, so things like exits being blocked by hazards, or any altercation to the scenario would not affect the simulation, hence not given a realistic result in all the possible scenarios as emergent evaluations are not present.
- **Cellular Automata:** A scenario is split into a grid, each grid-node has different states such as empty, occupied by human, occupied by obstacle, or a wall. Only individuals can move to an empty node, but they have a limited number of moves. One of its biggest problems it is that breaks the available space in an orderly manner and not letting individuals be on the same spot at once, these clearly does not reflect realistic behaviours as crowd will likely tussle for space as nervousness increases in an evacuation.
- **Agent-based:** Probably the best way to simulate crowd dynamics in the most possible realistic way, it is the approach taken by MASSIVE (Multiple Agent Simulation System in Virtual Environment) one the best-selling software in crowd-simulation effects. Their work is used in many blockbusters like The Lord of the Rings or even more recent movies like Avengers End Game [9] they achieve a realistic crowd simulation as each agent has its own attributes and an individual behaviour tree [14]. This same approach has been discussed by researchers where agents assess individually all the possible outcomes and can respond to emergent scenarios during an evacuation [15]. Agents’ moves are not limited like in the other two approaches, instead they follow a physical model proposed by Helbing at al.’s Social Forced Mode [16], arguably the core idea for this approach suggesting that agents are driven by their own motivations, presenting different forces that would move them at their desired velocity but deflecting from potential obstacles.

Multi-Agent-Systems (MAS) is another approach to model crowd dynamics, it is related to Agent-based models, however this system can be quite complex, and it is capable to teach agents tackle emergent problems during evacuations, in fact that is what Lisa Torrey from St. Lawrence University proposes in her paper about Crowd Simulation Via Multi-Agent Reinforcement Learning [17], she suggested that agents could learn through RL(Reinforcement Learning) via trial-and-error, when an action is performed right a reward is given, updating the agents knowledge database. She proposes to use an algorithm that could be proven effective, Q-learning by R. Sutton and G. Barton [13]:

$$Q(s,a) \leftarrow r + \ max_{a'} Q(s',a') \quad (1)$$

$Q(s,a)$ where a is the action performed by the agent and s the current state of the environment, the Q value assesses the maximum possible reward, adjusted after every iteration starting from zero, where r and s' are the reward and consequent state after carrying out actions in s, and finally a' ranges all actions [17]. Torrey points that agents are designed to get a reward regarding their own believes and motivations.

Agent-based actions are very extendable, Almeida, Rosseti and Leça Coelho [6] make the most by using MAS in their framework, they propose different attributes that helps agents to evaluate different situations, these are: agents' health state ranging from healthy, injured, to dead, their mobility status with a veering speed depending on their panic state. They add vision to their agents so they can detect obstacles, the vision range is affected by their health, however they do not mention if the vision will be reduced by environment factors, like smoke. Agents have also a reaction time consisting on a delay to take the next decision. They introduce sociology factors like collaborating with other agents and their persistence to follow an evacuation plan based on how efficient it is, this last attribute its being constantly re-calculated. Finally, they consider other factors of the agents such as its knowledge about of the surroundings, gender, age, previous experiences in similar events, nervousness, and its role in the evacuation.

Another piece of excellent work that demonstrates this model's potential is the framework developed by Guy, S., Curtis, S., Chhugani, J. and Dubey, P., PLEdestrians: A Least-Effort Approach to Crowd Simulation [8], this framework acknowledges that individuals try to reach their destinations with the less amount of possible effort. They propose the following formulations:

- An energy formulation to calculate individuals' metabolic energy use when traversing,
- An effective trajectory computation minimizing the overall use of energy for a trajectory, while avoiding collisions with obstacles.

These formulations achieve a great result by generating random occurrences in crowds being collision-free. The principle of PLEdestrians is that individuals will choose a path based on the required effort to reach their goal, prioritizing shorter paths at a realistic desired pace. Each individual has a circle around with a given radius for collision avoidance and computes a range of permissible velocities. These velocities will be updated in correlation to the energy consumed by individuals during the simulation, meaning that if a desired speed requires a huge amount of energy the agent will then prioritize a less metabolic consumption rather than such velocity.

Individual's biomechanical energy is calculated in Joules (J), measuring oxygen consumed during locomotion as:

$$P = e_s + e_w |v|^2 \quad (2)$$

"where $|v|^2$ is the instantaneous velocity, and e_s (measured in J/Kg/s) and e_w (measured in Js/Kg/m2) are per-agent constants." [8] with this formula the total energy expended by walking is calculated:

$$E = m \int (e_s + e_w|v|^2)dt \quad (3)$$

Where m is the individual's mass. PLEdestrians is capable of computing a trajectory for each agent that is energy-efficient, whilst avoiding collision with obstacles. The algorithm runs in O(n) time, where n is the neighbouring obstacles used to calculate non-colliding restraints.

Even though the PLEdestrians simulate realistic scenarios, such as jams, arch-shape formations, lane formations, wake effects, etc., it has some limitations as the framework is only based in walking at normal speeds, they are not capable of modelling running or representing panic accurately. It ignores behaviours that could emerge in an emergency such as a stampede effect, which will slow agents as individuals fall creating new obstacles or squeezing of individuals at narrow corridors.

Newer research from T. Liu, Z. Liu, M. Ma, T. Chen, C. Liu and Y. Chai [19] proposes a crowd simulation based on the Funge's cognitive modelling [20], according to it an individual can be split into four layers (from bottom to top): geometric model, inverse kinematics, physical model, biomechanical model. Funge's states that cognitive models go further than behavioural models as the former can control what an individual knows, how the information is acquired and how its processed to take actions. With this cognitive model researches propose the components used to model individuals [19]:

- Sensors: these collects and responds to scenario's emergent information.
- Perceptrons: these differ from sensors as they are used to manage and choose sensors' information.
- Cognition: one of the most complex components of an individual, it contains belief, desires, motivations and social attributes. Depending on these a desired route of action is taken, and emotions are sent to the actuator's component.
- Actuators: this will change the attitude of an individuals based on the current individual's emotion.
- Body data: the geometry model.
- Physiological: just like in previous works, researches consider keeping updated a health status throughout the whole simulation. An interesting remark is that if the individual's health is close to 0 it could go into an unconscious status, something not discussed in previous works that could lead to new interesting result regarding to individuals responding to other unconscious individuals.

The crowd modelling also proposes a personality model based on the OCEAN model [21], divided in five factors,

- Openness: Creative aspect of an individual.
- Conscientiousness: Carefulness aspect of an individual.
- Extroversion: If an individual is extroverted or introverted.
- Agreeableness: How likeable is an individual and its behaviour towards others.
- Neuroticism: The instability factor of an individual and its negative emotions.

Each of these factors have a positive and negative value, the following table describes with adjectives each value:

O+	Curious, alert, informed, perceptive.
O-	Simple, narrow, ignorant.
C+	Persistent, orderly, predictable, dependable, prompt.
C-	Messy, careless, rude, changeable.
E+	Social, active, assertive, dominant, energetic.
E-	Distant, unsocial, lethargic, vigorous, shy.
A+	Cooperative, tolerant, patient, kind.
A-	Bossy, negative, contrary, stubborn, harsh.
N+	Oversensitive, fearful, dependent, submissive, unconfident.
N-	Calm, independent, confident.

(Table 1. OCEAN personality traits. [21])

From Table 1 we can deduce that individuals with higher chance of survival would be those with O+, C+, E+, A+ and N- personality traits, capable of reacting to changes and making a better judgment when comes to decision making. Those with the opposite traits will likely not survive if the situation gets severe as they are more prone to react with extreme behaviours.

During an evacuation these personality factors will have a major impact on the individuals' role [19]:

- O+, C+, E+, A+ and N-: are more likely to assume a leader's role.
- O+, C+, E+, A+ and N+: are more likely to be irrational during an evacuation.
- O+ and N–: are more likely to calmly follow the leader, but judge whether the decision making of the leader is rational.
- O– and N+: are more likely to blindly follow the leader, being dependant from others.

Researches acknowledge that during an evacuation individuals' psychology is constantly evolving with the scenario. Two interesting constants are presented, fear and hope. Where fear is related to increasing worry regarding the individuals' location from an exit, the magnitude of the event, and what they can observe from others reaction. And hope is linked with the individual’s location from a safe zone, reducing the intensity of fear. [19]

The individual modelling in this research excels at simulating a social model however it lacks from a more centralised crowd approach, not discussing various events that could happen where multiple individuals are gathered, such as flocking, instead in this research tests are with a few agents in an hospital, an unrealistic setting as hospitals tend to be crowded.

2.5. Current Software

Crowd simulation software is currently available either as plugin for other graphics engines such as Maya or Unity, or as standalone products. However, most of them are widely used in filmmaking rather than for the purpose of this paper, products like Miarmy[22], or Golaem[23], both Maya plugins are mainly developed to create astonishing battles, filling stadiums, or to simulate a large amount of entities for TV shows or movies, these software allow the industry to create a more realistic setting for their creations and save them costs in hiring extras. MASSIVE [14], briefly introduced earlier, is another crowd simulation software, that is used for the same purpose as the other two plugins mentioned, yet it is also used in architectural visualization, it can populate a scene with many agents that are able to traverse through the environment. Nonetheless these could not be used for simulating evacuations as they are not programmed to simulate the behaviours described in this paper, rather they just have a locomotion system with a desired path set before the simulation. Oasys[24] is a standalone software that simulates crowds used for urban planning and testing evacuation plans, by analysing crowd flow rates. Finally, uCrowds[25], is a plugin developed to be used in Unity that can be configured to simulated emergency plans or analyse crowd flows in a scenario.

It seems that all these products do not consider a psychological viewpoint, agents’ behaviour is either precomputed or very simplistic. There also seems to be a lack of crowd simulation software for emergency planning, my assumption from what I have been able to gather is that developing crowd software could be quite expensive and the current software is already reliable, also there may not be a huge market that could justify the high expenses.

2.6. Conclusions

Individuals just do not simple panic when an abnormal situation occurs, they may not even panic at all - however, they reason and analyse the situation from a rational perspective and to the best of their knowledge. Individuals will take different roles based on their personality, and they will follow others to safety and help those in need. Their behaviour will constantly change, as danger approaches individuals will try to rush to exits but doing so will slow them down by clogging exists and squeezing others. It is key that all these behaviours are accurately simulated thereby we can help emergency and urban planners who try to minimize loss of lives during an evacuation. Other methods to study crowds have been proven inefficient, therefore, this paper motivation is to research, improve current work on simulators, and develop a framework to understand crowd behaviour.

3. Research Methodology

3.1. Introduction

From the information gathered in the literature review, the required functionality for a crowd simulation was collected, aimed to be used for mimicking real scenarios. Having understood the current models for reproducing individuals' behaviour in crowds, I was able to develop a prototype.

This prototype was built using Unity3D, a game engine widely used in the industry for developing graphic interactive applications. It was also chosen by having a well written documentation online, providing all of the components required for this simulations and having an excellent backend that speeds up the development process as many of the elements required to produce interactive graphics demos are already built in the engine. In addition to our previous personal experience with the engine. At the time of the start of the development the project specifically made use of its latest version, Unity 2019.0.6f1.

All the scripts for the crowd's behaviour are original to this prototype and were coded using the programming language C#. The assets to draw the scene were either primitive shapes or obtained from Unity's asset store, those being:

- Snaps Prototype | Office by Asset Store Originals. Chosen for containing walls, tiles and pillars done in a modular way and with a low poly count that suited the prototype's needs to build the demo environment.
- Unity Particles Pack by Unity Technologies. This pack contains the fire visualisation used in this prototype, chosen by being reliable as it was done by Unity ensuring its optimisation.

Due to the time scale, using the Unity Engine, these ready assets and primitives played an important role at speeding up the progress of the actual crowd behaviour development, not having to worry about spending time on the development of 3D assets or a backend to render the simulation, which is out of the scope of this study.

3.2. Simulation Environment

The environment, illustrated in figure 3, is the default layout used in this simulation, spanning an area of roughly 52360 m^2. This provided a large area for the study of the crowd flow, with different layouts to explore different scenarios.

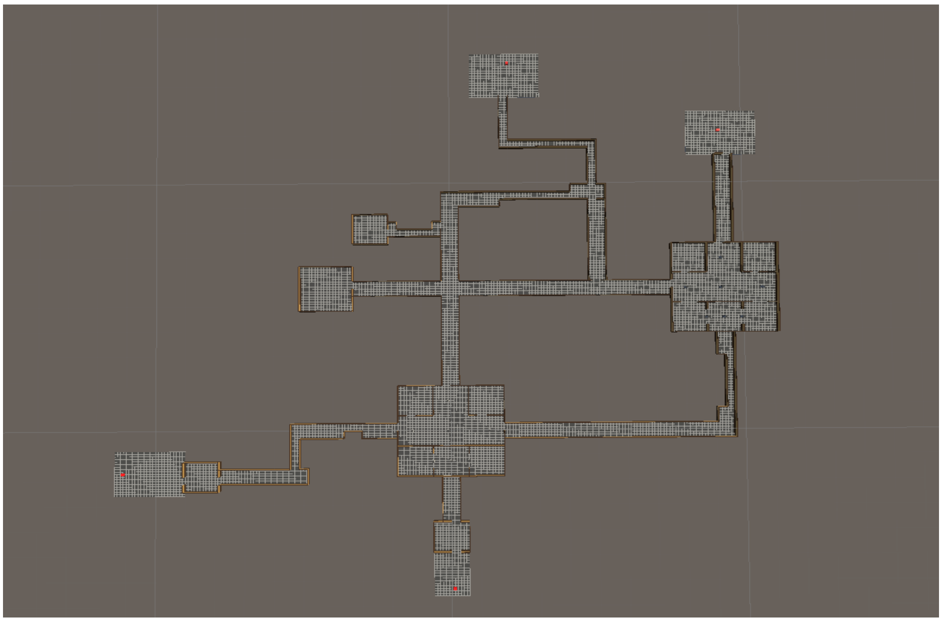

Figure 3. Layout used for the simulations

The structural design of the demo building was divided in 5 sections and they were done to research the following phenomena:

- Big Rooms: Done for examining the crowd behaviour in big areas, where pillars are present to analyse the impact of them in the crowd's flow.
- Medium and Small Room: Open areas where cramping is studied as the crowd tries to leave them.
- Long wide corridors: These are used to observe a wide enough space for a crowd to fill whilst traversing it and how the flow its affected.
- Small corridors: Here we observe how the crowd gets cramped whilst trying to rush to an exit, due to the small available space, unable to fit all.
- Loop corridors: Used to study the impact on the overall time. As crowds will loop back to the entry corridor.

It contains 2 **big rooms**, each containing different pillars to reduce overcrowding and to split those who are moving in opposite directions, those obstacles help the crowd's flow, these has been proven in different studies to reduce the overall speed of the individuals subsequentially having a good effect on avoiding cramping, and improving the overall escape velocity of the crowd by stabilizing its flow [26].

In the default layout **smaller and medium rooms** are created without pillars so the simulator can prove the obstacle's effect on the movement aligning with the findings of previous research.

These migration from room to room were done to demonstrate, as pointed in the literature review, that when a crowd panic and tries to go through a door, it may cause an arch-clogging effect, agents in the simulation act with the same panic described in previous studies pushing others trying to get out, increasing the percentages of flocking as the space available is reduced and the speed gets compromised due to constant forces. This effect can also occur changing corridors.

So , in the centre the demo contains a **long corridor** that creates a crossing connecting the big rooms with smaller rooms and a path to another exit; another long corridor on the east that connects exclusively the two main rooms; and four different **exits** distributed in the edges of the map.

In addition, there is a **small corridor**, on the north side, that leads to the main corridor but reconnects to it in the other end, on that small corridor another one branches off leading to an exit. However, this exit is rarely known by any evacuee, leading to interesting situations where evacuees may be near an exit but because of not knowing its locations tend to do a U-turn believing that they have reached a dead-end corridor. Proving the impact on the escape time, by those agents that may get lost due to either not getting guided properly or a bad structural design.

To simulate an emergency this simulation uses fire. For those the end-users two options are given - whether to start the fire randomly or choose the starting floor tile, in addition for another option to select the fire spread rate. The fire spreads to its neighbour tiles, and consequentially the more tiles affected the faster it spreads, ultimately blocking the path to exits and trapping evacuees.

There is also a Simulation Controller that controls the number of evacuees that needs to be spawned, and where. The areas can be designed through the editor to indicate where agents should be spawned. The controller also assigns randomly the leaders.

As an enhancement for those who are running the simulation time can be controlled, either speeding it up or bringing it to a halt. Finally, a camera controller provides free movement around the scene, so it is easier for the end-user to move around the scenario observing the crowd’s performance.

3.3. Agents Modelling

This prototype simulates the behaviour of each agent by using three different models discussed in the literature review, these being the Follow-Me-Method; Follow-Direction-Method; and the O.C.E.A.N personality model [9] and [21]. The simulations presented in this paper all run with 1000 agents. These agents are modelled through the same parameters regardless of their role, or model being used, these are:

- Age: The age of the individual that is selected randomly at the start of the simulation. Those who are under 16 will be dependent on other entity that serves as their guardian, one cannot leave or move to different directions without the other.
- Speed: The velocity of every agent is affected by the age factor. Those that are considered elder, 65 onwards, experience a speed reduction.
- Awareness: Perception of the events happening around the agent and the presence of other entities.

These parameters are chosen based on the information gathered in the literature review, these help in giving the simulations a more diverse environment. All of these parameters were selected because they rely on velocity, these are key to modify the overall flow of the crowd without being monotonous but adding variety. These three parameters are computed forming a final velocity that will affect the average speed and escape times in the evacuation. This adds different results based on a generation where all entities have different parameters from the last simulation. Also by introducing the concept of not leaving family member behind, in this case a child and a parent, we see an additional factor that impacts velocity, if one of the members falls behind the other will have to stop, impacting the performance of both entities. As we observe the phenomena discussed in [1] where family members will try to stick together regardless of the situation. Regarding the age groups, we assume that elders are slower based on a macro scale according to the literature review, although not all elders share the same movement speed, those who are closer to the adult age will be faster.

Then each model expands the agents' parameters, yet, both Follow Methods share a common parameter which is the leader trait. A leader is designated randomly at the start of the simulation, the number of leaders per simulation can be selected in the simulation controller's settings, in our demo only 2 are generated to do an stress test on their capabilities, and if they can handle a thousand evacuees. Leaders are divided into two groups:

- "True leaders" who are provided with the exact location of the safe exits to guide safely the evacuees out.
- **"**Rogue leaders" that may not know the right exit and guide the evacuees through the wrong path leading to chaos, this is shown in the later results.

The reasoning behind choosing the leaders randomly is the research in the Follow-Methods are not based in a complex psychological model, like the O.C.E.A.N model. But rather these two Follow-Methods take a more simplistic approach as the objective for these models is to simulate the crowd's flow when the agents are being guided to an exit, or struggle to find guidance, and thus the agents are simulated as so in this prototype.

In the Follow-Me-Method the leaders instruct the agents around to follow them to the exit, although the evacuees are not aware of the actual location of the exit. And they are only instructed to follow the leader to get to safety, those who see the leader and are also lost join the leader to the exit, but those who are already in a group or already heading towards an exit will ignore instructions. This means that agents are intelligent enough to ignore contradictory instructions when a supposed safe exit has been given to them.

In the Follow-Direction-Method the leaders instruct those around them the direction of the exit, and then proceed to the exit themselves. So, the evacuees know at all time, once being instructed, where the exit is, therefore, no longer requiring having the leader in sight to escape.

In both Follow methods, those evacuees who are not aware of the exit, would try to follow the same path that they used to enter the building, until they find a group of evacuees being guided to the exit. Other evacuees may not know their way out due to not knowing the layout of the building so they may end up going to a dead-end corridor or take the wrong turn when it comes to a T cross. This behaviour is like the one described in the literature review, regarding evacuees relying on their instincts to try to get out of an emergency. In addition, agents are able to adapt to the environment by reacting to fire and running away from it. This happens when they see the fire and acknowledge that the exit that they were trying to access is no longer available, hence trying to head to another, and proving that they are instinctively reacting to the emergency.

The O.C.E.A.N model implemented in this simulation supports four personality groups, which are, as described in previous research [19] [21]:

- O+, C+, E+, A+ and N-: Agents assume the leadership role, they will know where an exit is, and will guide others to that exit. Their role is assigned due to them, according to Table 1, being alerted, informed, having perspective of the environment, persistent, energetic, calm, independent and confident. They have the adjectives that defines a leader in an evacuation, and thus look at by other entities.
- O+, C+, E+, A+ and N+: Similar to those with leadership traits, but the major difference that make then completely opposite is the N+, this means that agents with this factor will be negative, fearful, unconfident, and most likely to act irrationally, therefore in the simulation their attitude will be highly volatile, like not following indications by leaders, and tend to easily give up.
- Following personalities, these will stick near the leaders and follow them to the exit, simulating herd behaviour, however there is a distinction between two different kind of followers:
    - O+ and N-: calm agents that will follow the leaders indication, but always doubtful about its decisions, if they deduce that the agent is heading to a non-safe area they will break from the group and try to head to another direction.
    - O+ and N+: like the agents described above but will blindly follow the leader without questioning its decisions.

Agents are placed into one of these personalities group at the beginning of the simulation with higher chances for them being assigned into one of the Following groups, and a lower probability for irrational and leadership personalities, as these are considered uncommon. The speed of the adult agents are affected depending on their N trait, if they fall in the N+ the speed will be increased due to fearful component, we assume that they will try to rush to exits, and try to evacuate as soon as possible, breaking from the crowd's pace. Yet this can cause the unexpected output of freezing the overall speed of other evacuees due to the collisions that are caused with those going at a calmly pace, N-.

To move the agents around the environment the prototype makes use of Unity's NavMesh agent, a form of abstract data structure, that stores convex polygons and creates a graph of nodes (polygons) that define the traversable, "walkable", area of a map [27] that comes with the features required for this simulator. In addition, to navigate the agents we built a waypoint system, with keys locations on the map, those are crossings, intersections, main rooms, and waypoints that are tagged as exits. All waypoints have the option to be linked to another one, meaning that once the agent reaches the waypoint's destination, their next destination will be the linked waypoint. Exits have an additional parameter to indicate whether they are safe or not, this parameter is only known to the leaders, or those evacuees who reached the exit. Using Unity's built in navigation system provides the simulation with all the requirements for agents to intelligently traverse the map. Waypoints are used to enhances the agents' intelligence and what would be the expected movement through the map based on the main areas. This waypoint system is easily extendable, and waypoints can be added as required, giving more freedom to those who are running the simulations. Allowing them to create scenarios to analyse different case studies, like loop corridors.

The waypoints have a probability of being known by the evacuees, where rooms, corridor or exits, that are most likely to be unknown or not intuitive as the way out for those who are not familiarise with the building, have a lower percentage of being found or taken. Therefore, introducing a realistic scenario, as agents will most likely try not to go down that path, unless they are told by a leader, or other routes are blocked.

As the fire spreads the agents' traversable path will update, and recalculate a new route if the target exit or waypoint is unavailable, those who get completely blocked by the fire and not being able to evacuate will turn off their behaviour and deactivate, counting as trapped by fire.

From a technical standpoint, one of the biggest challenges behind simulations with a huge number of agents, with each containing their own update loop and independent behaviour, is performance. During the development, this project has always targeted a stable performance of 30 frames per second. Even though humans react around at an average of 250 milliseconds [33], being far slower in comparison to the entities in this simulation, realism is not really affected, nor the fact that agents react faster is noticeable. Technically they do, but all of them are updating every frame, so none of the agents is ahead of others, so they are still prompted to the same phenomena that occur in real life, that is why the objective of this simulator are to represent as many agents behaving individually and making it pleasant for the end-user to look at a simulation that does not jitter by achieving a stable performance.

To achieve so, various techniques are used, like pre-calculating the path that the agent needs to take before taking it, avoiding allocating memory when detecting other entities around, and instead storing the results in buffers, thus reducing the work done by the Garbage Collector. The overlap sphere detects which entities are around the agent, and stored them in an array, that it is later used to iterate through it and compare its behaviour with others or check around it, like whether a leader is around them. Likewise, for the leader to check if entities are around them and send them movement instructions.

By using an Object-Oriented approach, a base for all agents is provided, this base contains basic expandable functions to detect entities, assign waypoints, a core update loop and a basic initialisation, making it easy to extend the base system into more detailed models but sharing the same infrastructure.

The structure of each agents is divided in three, physics, navigation, and logic, each expand in sub-components containing the required information to bring to agents to live and make them behave as expected, a visual visualisation is shown in Figure 4.

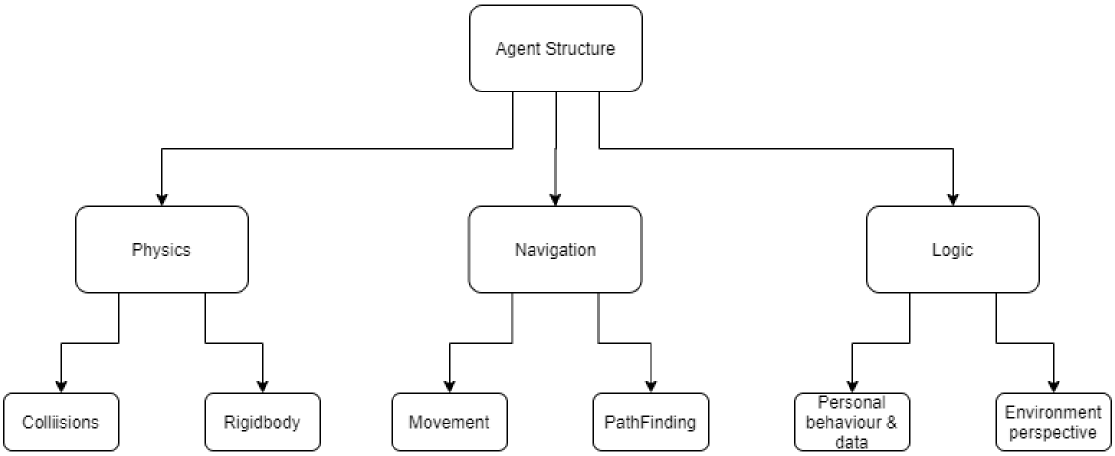

Figure 4. Agent Structure.

As an overview, the physics component oversees the collisions, making sure that the agent does not go through structures or others, with a Rigidbody it makes sure that this objects interact and allows the object to receives forces and apply real time physics, existence in the engine, Unity3d makes use by default of the NVIDIA PhysX physics engine, widely adopted by the industry and it is designed to use hardware acceleration, improving the performance of physics processing [31]. The navigation component takes care of calculating the agents' path and setting their destination to the exit, or to the leader. Finally the logic component takes care of storing the parameters that will define the behaviour of the agent, by using the overlap sphere, described earlier, where it gathers the entities around them and through them, performing actions according to the situation, and the behaviour of others around, adapting constantly to new changes in the scenario.

3.4. Data Generation Methods

During the simulation, data are collected for each entity, the time they start moving until they escape the building, unless they get trapped. This data is then exported into a CSV format that can be used in other applications like Microsoft Excel. It can be used to calculate means, generate graphs, or use the data at the disposal for any kind of data analysis, as well as comparations between different age groups. To store the data, we distribute the entities in different age groups:

- Children: Representing group of children, aged 15 or younger, with adults.
- Elder: Representing group of 65 and over.
- Adults: Representing all other ages.

Splitting the data collected in groups helps to identify those that may be slowing the overall evacuation speed. And the time taken to escape is also collected and represented in seconds. This data is generated at the end of the simulation and is spited in two sets.

A raw data containing the time that an agent took to escape from the building for every individual entity, stored in the following format, *Age group – Time Spent*. If the entity is not able to escape it will show as *"Unavailable to escape"*, instead of the time. And another data file containing the average escape times for all age groups formatted in *Age group – Average Time Spent.* A sample of the raw data it displayed in Figure 5 and the average escape by group in Figure 6.

Sensitivity: Internal

Age group	Time Spent (seconds)
Elder	244.5077
Adult	211.1483
Adult	177.4512
Adult	25.59669
Adult	230.3839
Child	62.72409

Figure 5. Data sample of individuals escape time sorted by age group.

Age group	Average escape time (seconds)
All	256.6477
Child	177.8111
Elder	257.5281
Adult	177.4512

Figure 6. Data sample of average escape time sorted by age group.

As observed in the sample figures the data is straightforward for any reader and providing insight to the performance of each entity and age group. In addition to distinguish those personality groups that perform better in the evacuation, the adjectives traits of each entity are included in the CSV of the first set of data.

Another form of data generation provided in this study is given in the form of live statistics.

The first being a visual representation, heatmaps. These are generated during the simulation, using a colour grading representing different values of agglomeration. When agents traverse the map, all floor tiles accumulate data regarding the unique agents stepping on them computing a colour scale based on the number of entities that have gone pass the tile. There are different configurable parameters given to the data analyser, like the maximum entities at max heat, and the minimum and maximum colour of the heatmap, used for the heatmap's hue, these creates a spectrum of both colours. Blending during the simulation, by normalising the current entities that have stepped on with the maximum until is fully heated. In the current model a scale of purple is used to represent the heatmap, and a button is provided to toggle it, enhancing the experience of the end-user by not obstructing its view.

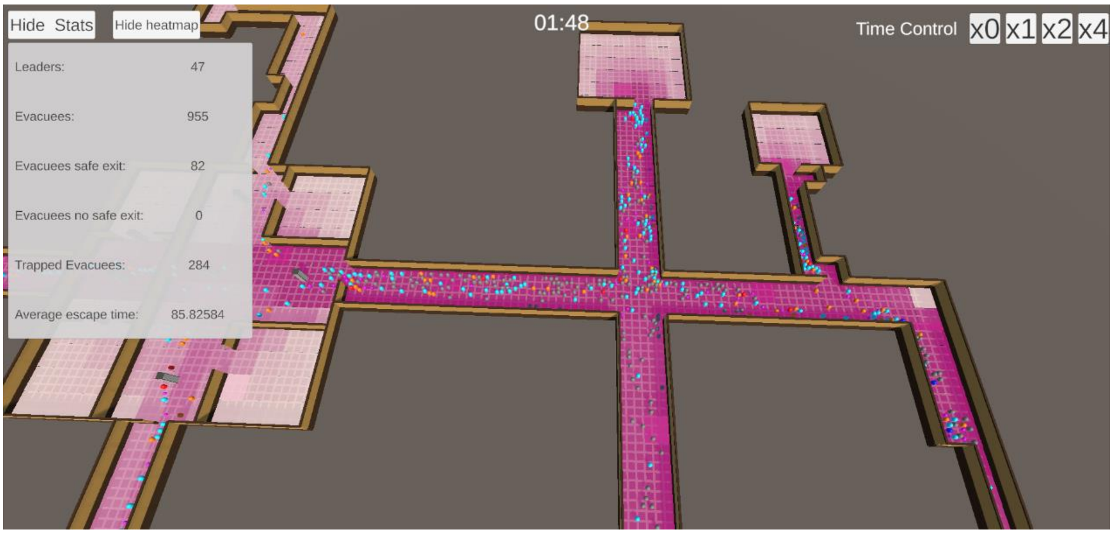

Figure 7. Simulator's live statistics and heatmap in action

Live statistics are also displayed whilst running the simulation. As the simulation unveils, data is constantly updated to give information like agents that have been trapped, or those who have escaped the building, or the evacuees' average time at that stage. Like with heatmaps, a button is provided to show or hide these statistics.

3.5. Limitations

The first evident limitation is how demanding are simulations with a large number of autonomous agents, and the more complexity the more weight it puts on the CPU. This problem can be addressed from its architectural root, as the prototype is using a data-oriented design, hence objects are allocated across all memory and data requires constant memory access, this is computationally expensive when running with a lot of objects that it is why the CPU tries to prefetch memory by caching it and then access it from faster memory, yet many Unity's components contain data that may not need proceeding taking a lot of space in the cache, ending up with a lot of wasted memory and consequentially having a toll in performance[28].

However, if it could be changed to an Entity Component System (ECS) [28] [29], where every element will contain just the right data that it needs to be process and those sharing the same component would be laid orderly together in memory, reducing the footprint of wasted memory in the cache, hence speeding up system performance. This pattern, as indicated in its name, splits objects in three: *entity* that serves as a tag or identifier; *component*, the entity's data; *system*, processes the information of the components. Compared to the standard Unity's Update loop, ECS takes advantage of the way it stores components, and linearly loops through them reading and writing data faster compared to the current approach [30]. Also, by making use of Unity's new Burst Compiler, it boosts performance by organizing memory based on the machine running the application, hence optimising the application around the capacities of the running machine [28] [30].

The simulator did not make use of this new pattern due to currently being in a preview stage and me not being familiarised with this new architecture. It was considered before the development, but due to time constraint and experience we decided to go for the classic Unity implementation, trying to optimise the code as much as possible and also serving as a proof of concept that could later be expanded into a more robust system by changing its architecture.

3.6. Conclusions

This section managed to introduce the demo environment used in the simulation, the different building structures employed for testing different crowd behaviours, and how these structures affect the overall outcome of the evacuation time. We explained the way agents were modelled. We used a multi-agent system approach to simulate individual's behaviour. Combined they created the crowd system seen in this prototype, using three different models allowed us to examine different phenomena observed in panicking crowds. These models were based on the literature review and were divided in two:

- The leader centric approach consisting of two Follow methods, with differences that created unique results. The Direction model where evacuees are told the exit's whereabouts by leaders, and the Me model, where entities are instructed to follow the leader.
- The other was a psychological approach that demonstrated the influence that the O.C.E.A.N personality traits have in crowds.

These three models provided diversity in the prototype, rather than focusing in one model we managed to deliver multiple options so the end-user could choose the model that better suits their needs.

We also introduced how the simulator collects data, these are either presented in form of live statistics, or gathering vital data, like age group and escape time in seconds, that later gets extracted into a CSV file to be analysed. CSV are easy to implement in code and expandable allowing us to append as many data as needed into a file, in addition it is supported by popular office software like Microsoft Excel, thus it seemed like a reasonable choice.

Finally, we reflected on the prototype's current limitations, but we also suggested a solution for the current performance problems that we faced. Future work could be related to migrating the current methodology to simulate agents into an Entity Component Base approach so buildings or evacuation plan where more than a thousand individuals would be expected can be simulated in our prototype. An object-oriented approach seemed reasonable due to the project's time scale.

4. Findings and Analysis

4.1. Introduction

In this section we provide a set of simulations and we compared the outcome with the literature review, so we can verify that the prototype works in accordance to previous research.

We use the simulations to describe the crowd behaviour, using pictures to visualise different events. Then, at the end we use CSV data generated and we extract it to be analysed. The data is obtained by running tests in the different models supported by the simulator.

4.2. Simulation Breakdown

4.2.1. Follow Me Model

Agents are distributed across the whole map, the fire breaks out in the north east corner of the building, and agents start making their way out to the exits. The leaders, responsible of the evacuation, start to indicate the way to those around them to move with them towards the exit. Within the first minute of the emergency only a few agents have made their way out; this is most likely due to its proximity to an exit. However, those who have not been able to escape from the fire are getting confined and for the next minute those that are unable to escape are trapped. In other parts of the map groups have already been formed and are approaching the exit. Congestions begin to appear in some turnings and corridors, slowing the agents. Those who are travelling in different directions and going through congestions, contribute to the reduction of the overall performance of the groups. After four-six minutes in, most of the groups have already found the escape exit and are evacuating calmly. Congestions begin slowly disappearing across the map, and lines are being formed towards the exit. It can be observed that a high number of elders are left behind as they could not keep up with the pace of the crowd. Others who are left behind are those who have not found the right exit or a group leader to guide them. Around the nine minutes mark most of the evacuees have evacuated the building safely. Yet, this simulation has been hindered because of the leaders' position far from the safe exit; it could have gone differently if they had been placed in another spot of the building closer to it. According to the findings from several runs of this model, the performance of the crowd increases as they have to traverse a shorter distance, meaning that they will encounter less entities travelling in different directions that may slow down the crowd, or have to go through less different corridors hence not creating congestion and increasing the overall social distancing, increasing the overall speed thanks to the less collisions between agents. The payoff is that it may leave a lot of entities without group as these will evacuate more quickly. Another problem that could take the wrong path, aggravating the situation as the path will be recalculated to the new safe exit, increasing the overall time to escape for a group.

4.2.2. Follow Direction Model

Like the other Follow model, at the begging of the simulation agents are scattered on the map. In the first minute, agents start making their way to the exits, or to the waypoints where they believe may be the right route to the exit. At the same time leaders are starting to tell those around them the direction to the exit that they must follow. Rather than group agglomerations start to form as evacuees head for the exit, cramping occurs on the corridors, and reduces the speed of the crowd, this creates an arch-clogging effect. In the peak of the evacuation most agents have managed to escape and head to the exit orderly. Like in the other follow direction those who are behind are mainly the elderly, in addition to those who have not encountered a leader that pointed them in the right directions. As in the Follow Me model the fire has trapped those who could not manage to escape. This model encounters a major threat, namely leader positioning: if the leaders are placed near an exit, they will jeopardize the escape time, as less evacuees will know where the exit is, due to leaders evacuating sooner. This would leave a lot of agents roaming around the map trying to look for an escape exit, and thus causing congestion as they collide with each other when travelling in different directions.

4.2.3. O.C.E.A.N Model

In this model entities are assigned to a personality group at the begging of the simulation. There is a higher probably for having more leaders in this simulation but also a higher probability of getting rogue leaders. That is why doubtful followers will look for alternatives when being guided to the wrong exit, creating a sub-group of their own that has broken from the main group. Like in the Follow models the agents start making their way to the exits, although the major factor is the difference in speed between the agents with N+ being faster and N- being slower, in addition to the elders' slower speed. This diversity in speed if congestion occurs it will be greater than the other models as the speed of those more prompted to panic will disturb the crowd flow when trying to get through them to get to the exit. But overall, the congestion in this model is usually smaller as there are more leaders guiding smaller groups to the exit. Like in the follow models, the last to evacuate are the elder, and the fire has similar consequences as other models.

4.3. Analysis

Based on the simulations, all models share a similar output, which is the effects described in the literature review regarding the crowd flow.

4.3.1. Clogging Effects

One of these is Arch-Clogging effects, the current map layout creates the effects described by Helbing [7]. These effects can be seen when agents collide with themselves and occupy as much area as possible in a corridor or room, not respecting any kind of social distancing. When it comes to changing corridors, or rooms, the effect occurs, due to the sudden change of available area. As the passage is smaller than the previous one, all of the agents will cramp on the transition zone; because of their collision component, they will not be able to go through all at once, yet each individual agent would try to push their way through. This constant pushing reduces the performance of the crowd, since they are not able to escape orderly, and not respecting the queue position in the group, the individuals of the crowd act selfishly and a few manage to break from the arch every second. Phenomena like freezing by heating also occurs, for instance when the path is completely blocked by agents, the crowd gets stuck and it stops moving completely. Also, another phenomenon described in [7], namely faster is slower, can be seen here: the false thinking of certain individuals that going faster will mean that they escape sooner slows other around them, and consequently slows them down.

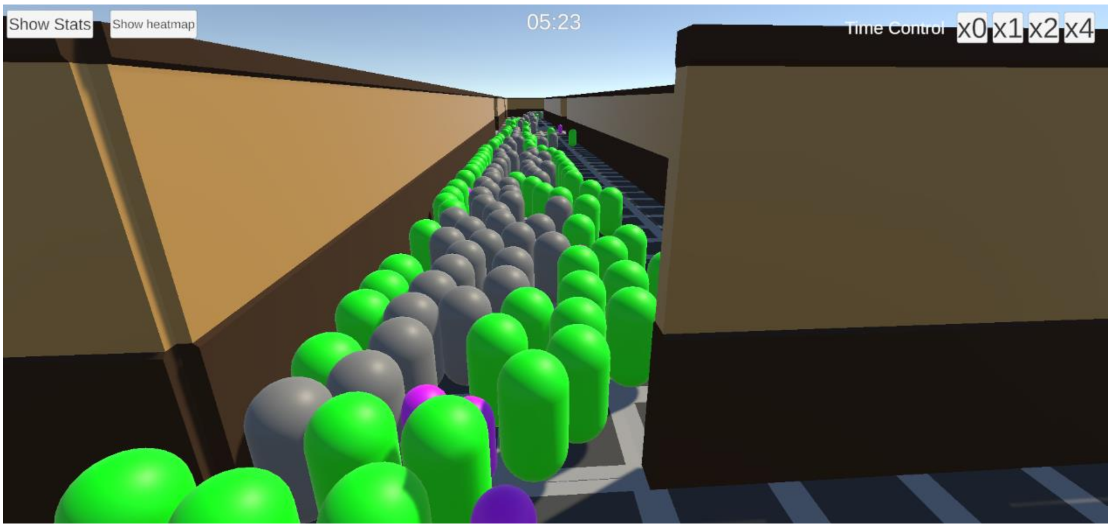

Figure 8. Arch being formed at an exit. From a close-up view.

4.3.2. Leader Influence

In the Follow methods the position of the two leaders influence the overall outcome of the simulations, the closer they are to exits the least evacuees they will drag with them to the exit, being ineffective in the simulation. On the contrary if the leaders are placed far from the exit, they will travel a greater distance, dragging a lot of evacuees with them causing a higher disruption as they will change directions when being instructed by the leader and become part of a larger gathering of entities all heading to the exit. The ideal scenario is when leaders are situated in the centre of the map, where they will keep the balance between congestion and reaching out to as many as possible.

4.3.3. Corridors impact

Small corridors are the biggest threats for crowds in a panic state: the smaller available area means the worst outcome for the overall movement speed. If we were to take into consideration health conditions, the pressure built upon the crowd when being flocked in a corridor will likely cause injuries, that is why studying these scenarios where blockage may occur is vital for researchers in order to improve evacuation plan or design. On the contrary, this helps to prove that pillars and larger areas help split the crowd flow where agents barely collide, or the velocity is not jeopardized. The heatmap shown in the simulation provides the areas where most agents congregate.

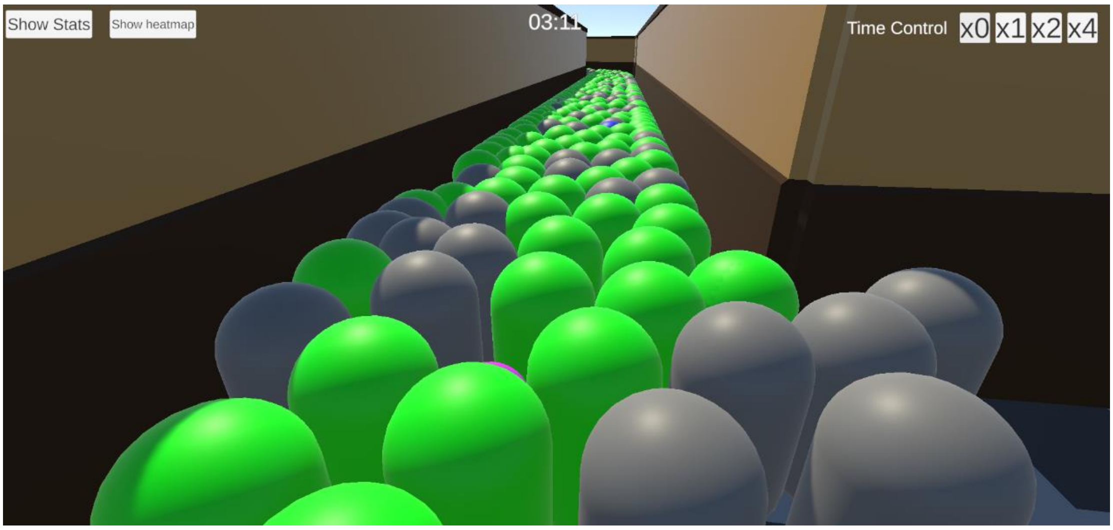

Figure 9. Cramping occurring in a small corridor.

4.3.4. Data analysis

From the CSV we extracted the data based on the model's simulation broken down earlier in this section. With the data we can compare the efficiency of each model and study the different output, advantages, and payoffs for the three different supported models. Nevertheless, bear in mind that each simulation is different to the last one due to the different factors that affect the crowd's escape time, these are:

- Leader positioning, this will have an impact whether the leader is close to multiple individuals, close to an exit or far from it. Increasing the escape time or reducing it.
- Fire spread: those exits that get blocked by fire means that entities will have to turn around and head to another exit.
- Clogging: because clogging effects are dynamic and may occur or not these will have an impact on the escape time.

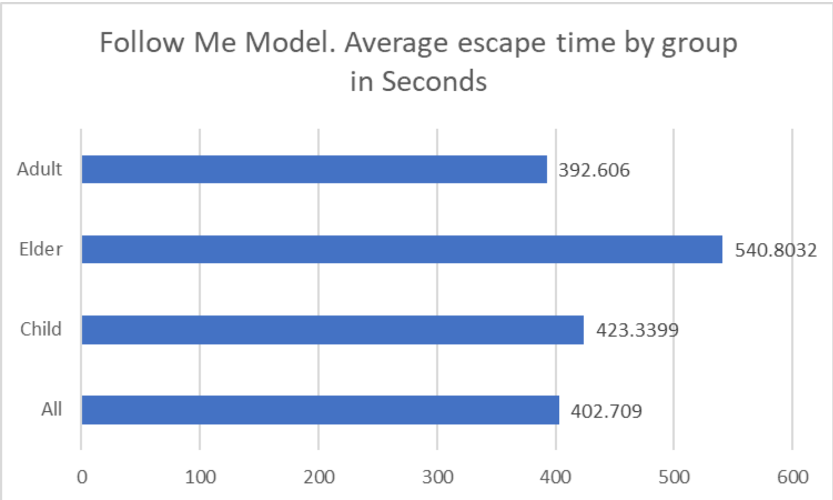

Figure 10. Follow Me Model average escape time

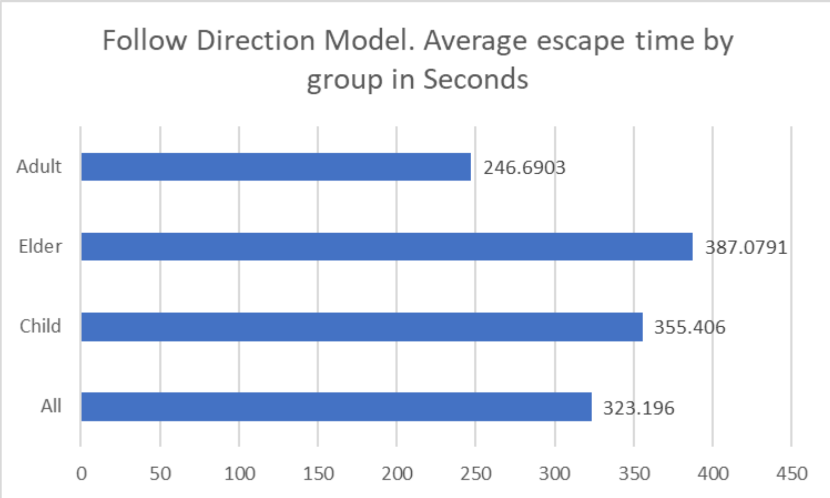

Figure 11. Follow Direction Model average escape time

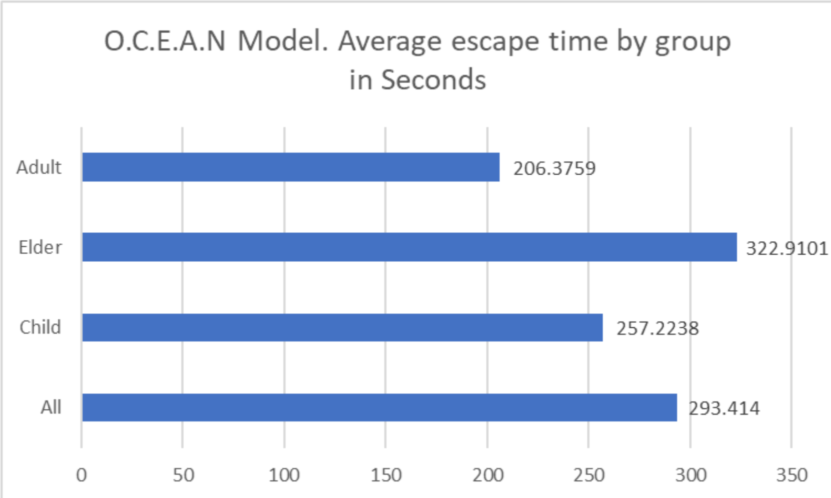

Figure 12. O.C.E.A.N Model average escape time

As seen in Figure 10 and explained in the simulation breakdown, the Follow Me simulation was unfortunate with its leader placing, even though it drew a larger number of entities, it was not enough, resulting in being outperformed by other models. Also, due to having to traverse a larger distance the collisions with other groups or entities reduced the escape time. Overall, it had the worst escape time for all age groups. Like in Murakami et al. [9] research the follow-me method with two leaders is the least effective, of the two follow methods. In this simulator is due to the distance being traversed, and in alignment with Murakami et al. the ineffectiveness of two leaders having to manage a large group of evacuees.

The Follow Direction model gives better results in the simulator compared to the other fellow method, as seen in Figure 11. The position of the leaders was favourable as they were placed in centric parts of the building, indicating to the right number of evacuees the exit's position, and not generating a big agglomeration as they traversed the environment. It achieved the best adult performance.

However the best performance overall was the O.C.E.A.N model, this is due to the large number of leaders that were generated during the simulation, these were able to guide in smaller groups evacuees to the exit, and due to a higher calm percentage of individuals it achieved a simulation without any major problems apart from smaller congestions at certain tight corridors.

By looking at the raw data, most of the evacuations take place between the 200-400 seconds, these patterns repeat in the three different models. According to research done by Doctor Proulx, individuals will take some time to react to the emergency [4], and this is clearly seen at the beginning of the simulations. The target for evacuations should be 2.5 minutes, yet according to a survey by insurer Royal Sun Alliance [32], people may take up to a minute before deciding to leave, and that it is why this simulator may help urban planners or evacuation strategists to situate exits and adjust the number of leaders, if they decide to go for a Follow model, to hit that target evacuation mark, by taking into account the delay that may occur, before evacuating. We also need to consider that only two leaders oversaw the evacuation, and yet they achieved a satisfactory result considering the large amount of entities that they had to guide. Increasing the number of leaders, like in the O.C.E.A.N model were 43 were randomly generated with the leader personality trait, could boost up the performance of both Follow methods.

These analysis examples show how an end-user could benefit from the data extracted from this simulator. They provide diverse results that may help the researches to come up with an average for the three models and detect the similarities occurring across all, so they can fix the weak points in the evacuation.

4.4. Conclusions

The analysis of the simulations run in this section prove that the prototype aligns with the data and research done previously. By using a multi-agent system, compared to other architectures, we were able to shape each individual agent allowing them to take decisions individually but with consequences for the bigger group, the crowd, if a decision is poorly made. In this section we explained the different simulation breakdowns, and how this are very similar to each other but with different aspects that influences the performance of the crowd, such as showing all the effects that a panicking crowd may go through in an emergency.

The analysis has proven that the key factor in simulations regardless of age, race, gender, being around with family or psychological traits, is speed. Maintaining a good pace during an evacuation is determinant to avoid occurrences that will put in danger the security of the crowd. Keeping a velocity target that can be achieved by all crowd members is vital, going too slow or going too fast is not the solution, the false belief of thinking that going faster is equivalent to getting out of a dangerous situation is a rational instinct typical of humans. And the unfortunate effects of the aging process in the human body makes elderly more prompted to have a reduced speed capability. Both result in a slower crowd flow, by pushing agents out of their way and consequentially creating a bottleneck. To them it may seem rational to keep a tranquil attitude when the end could be near, it is survival what drives these behaviours, and to us it may seem irrational, as external examiners, when we see that the panic in the crowd is generating more problems and weakening its pace. That is why that we need to take into account these aspects in our simulator, and its subsequent analysis, so we can improve the security of the crowd and ensure that the right measures are put in place, in order to try and keep a calmly pace during the evacuation and be aware of the different patterns in velocity so we can split them or allow enough space for the crowd to head safely to the exit.

5. Conclusions and Recommendations

5.1. Conclusions

This study provided an extended survey in previous research done in the field. Observing the different patterns that occur in a crowd when they enter in a panic state, the behaviour of individuals within a crowd and the factors that affect the final evacuation time. We also had a look to existent software and simulators from different research. One of the problems that some of these simulators or software faced at the time of the study was the constraints on the hardware, the ability to simulate thousands of entities like this study does by using a multi-agent system, where the fact that agents behave autonomously was a huge load on the CPU. Having to rely on other architectures that do not provide as much depth in simulating multitudes of individuals, as cellular automata or particle-based simulations do. But this study at the time has been favoured by the improvement of technology both in hardware of software over the past decade, having been able to run an expensive graphics crowd simulation at a stable rate of 30 frames per second. The three models represent different behaviours that the end-user running can choose from and analyse the output. Other tools provided for the analysis of data were provided, like heatmaps, and collection of data to be exported to a CSV file.

The follow models based on the ones described in the literature review provide an overview and an approach for simulating a behaviour that it is based in a leader centric approach. And the O.C.E.A.N model shows the approach from a psychological standpoint. These models allow end-users to analyse different evacuation patterns, hence achieving the objective set by this study, namely, to create a simulator where urban-planner, emergency planner or others that may be interesting in crowd mechanics benefit from the simulator.

The study achieved a simulator that can reproduce real events, where different study cases can be run by those using the simulator. However, this study failed to provide a documentation for the end user to setup the simulator. Although this would come at ease, as the simulator provides a very intuitive set up. A temporal alternative solution could be replacing the existing demo layout with a new building layout. Yet, this is not ideal and the next step, regarding to future work, would be creating documentation, tutorials, and useful resources related to get the simulator up and running and for those who may not be familiar with Unity and the systems within it. The idea would be to extract the prototype as an addon to be used within the engine. The reasoning behind not having a standalone release is that it would require additional work regarding the creation of a 3D editor and an in-built configuration. By using the Unity editor, already optimised as it has been through different iterations throughout the years. Besides using the inspector where all the scripts' public variables can be modified instantly, and at runtime. Additionally, these tools helped the prototype to centre the attention and all of the development effort around the things that mattered the most, modelling the behaviour of agents in an emergency.

Yet after this study we understand why the market for simulators is not that attractive for larger enterprises, the market is already suited with simulators that do the job, and can generate simulations that suits their needs, regardless of how complex they are from a psychological perspective. It is also expensive and time consuming to develop a crowd simulator that is faithful to real crowd interactions, the amount of power and optimisations needed to make it run with fidelity to real situations can put away newcomers. Most of the current simulators just deliver an evacuation stress tests, that may be what the urban planners are after, using it to check the different weak points of a plan or a building layout.

Nevertheless, the motivation behind this study was to investigate further into crowd behaviour, the benefits of using an approach that is not based in a simple particle advancing to an exit, but taking it into a step further of simulating real occurrences, examining the human instincts and factors that affect an evacuation to be simulated aligning as close as possible reality. This simulator has achieved so, yet it could have gone more in depth into the psychological research, many aspects has been simplified due to the scale of the project. Future work in this projects includes simulating behaviour with disable entities, to study how well adopted are the evacuation plans for those who may have a disability, like being in a wheelchair, and adding a factor of health, to test for injuries when cramping and how would they affect the crowd flow, most likely resulting in a speed reduction. These features could not be added to the simulator in this small development time scale.

Responding to the question of the title of this study, disasters are uncontrollable and unpredictable. Unfortunately, they may happen and those who oversee our security should make use of the tools that are available to make evacuation safer. I believe that this study with its simulator proved how planning and running different case scenarios can help in predicting different crowd behaviours, like agent congestions and weak points in an evacuation. All of these can be used to our advantage to improve our security, not just in buildings, where they are already heavily subjected to rules regarding to fire safety, but also in open areas where big gatherings are formed. These could prove useful to prevent stampedes in case panic unleashes. To conclude that our simulator can provide insights in the dynamics of crowd, whose complexity involves challenging aspects of human behaviour, sometimes unpredictable. And answering to the question raised on this study, yes, our simulator can save lives as and can improve our security. Even human behaviour can be challenging and unpredictable sometimes but by using model like the O.C.E.A.N the simulator tried to get a close approach to reality, and predict the outcome of an evacuation, ultimately helping to improve our security.

5.2. Recommendations and Final Thoughts

This study recommends future researchers to invest time in making the next simulators using an Entity Component System approach to reproduce agents' behaviours. This new solution supported by engines like Unity boosts performance to the next level: it will allow developers to instantiate more entities and increase the behaviours' complexity. A more complex system will allow researchers to analyse crowds deeper and come up with better solutions to existing problems in evacuations. Improvements to the current models could be done in regard to simulating the small percentage of entities that are likely to have an underlying health condition, examining how these conditions may affect the evacuation outcome.

We also encourage to push the research into the market and showing its value in the industry of urban planning and safety. These simulations, with the right amount of complexity and detail can make a difference in the world of safety, improving our security by finding and detecting the problems that a planned project, or an actual evacuation strategy, may face when applied to a real world scenario.

Finally, this research could also be extended in the context of games: indeed, certain videogames could benefit of realistic crowd behaviour, helping players to engage with the crowd and feel immersed in the world. Crowds could react to events in the game, realistically improving the game’s narrative and serving as a selling point. Other possibilities include porting crowd systems to Virtual Reality, serving as a study to experience the feeling of being in a crowd during an evacuation, experiencing the panic and the different phenomena described in this study.

6. References

[1] A. Mawson, "Understanding Mass Panic and Other Collective Responses to Threat and Disaster", *Psychiatry: Interpersonal and Biological Processes*, vol. 68, no. 2, pp. 95-113, 2005. Available: http://www.csap.cam.ac.uk/media/uploads/files/1/mawson-2005-emotional-attachments.pdf. [Accessed 4 December 2019].

[2] BBC, "How the Hillsborough disaster unfolded", 2016.

[3] J. Keating, "The myth of panic", *Fire journal*, vol. 76, pp. 57-61, 1982. [Accessed 6 December 2019].

[4] G. Proulx, "Occupant Behaviour and Evacuation", *Fire Risk Management Program*, 2001. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.11.202&rep=rep1&type=pdf. [Accessed 5 December 2019].

[5] D. Helbing, I. Farkas, P. Molnár and T. Vicsek, "Simulation of Pedestrian Crowds in Normal and Evacuation Situations", 2008. Available: https://pdfs.semanticscholar.org/6b72/b31c2531b2a910ecbe6baa095b0907a7448a.pdf. [Accessed 7 December 2019].

[6] J. Almeida, R. Rossetti and A. Coelho, "Crowd Simulation Modeling Applied to Emergency and Evacuation Simulations using Multi-Agent Systems", 2013. Available: https://www.researchgate.net/publication/228610829_Crowd_Simulation_Modeling_Applied_to_Emergency_and_Evacuation_Simulations_using_Multi-Agent_Systems. [Accessed 10 December 2019].

[7] D. Helbing, I. Farkas and T. Vicsek, "Simulating dynamical features of escape panic", *Nature*, vol. 407, no. 6803, pp. 487-490, 2000. Available: http://angel.elte.hu/pedsim/pdf/panic.pdf. [Accessed 10 December 2019].

[8] S. Guy, S. Curtis, J. Chhugani and P. Dubey, "PLEdestrians: A Least-Effort Approach to Crowd Simulation", *Proceedings of the 2010 Eurographics/ACM SIGGRAPH*, 2010. Available: https://www.researchgate.net/publication/220789236_PLEdestrians_A_Least-Effort_Approach_to_Crowd_Simulation. [Accessed 10 December 2019].

[9] Y. Murakami, K. Minami, T. Kawasoe and T. Ishida, "Multi-agent simulation for crisis management", *Proceedings. IEEE Workshop on Knowledge Media Networking*, 2002. Available: https://www.researchgate.net/publication/3981832_Multi-Agent_Simulation_for_Crisis_Management. [Accessed 10 December 2019].

[10] F. Vicente Castro and J. Pabico, "Microsimulations of Arching, Clogging, and Bursty Exit Phenomena in Crowd Dynamics", 2019. Available: https://www.researchgate.net/publication/279309876_Microsimulations_of_Arching_

Clogging_and_Bursty_Exit_Phenomena_in_Crowd_Dynamics. [Accessed 10 December 2019].

[11] G. Santos and B. Aguirre, "A critical review of emergency evacuation simulation models", 2004. Available: http://udspace.udel.edu/bitstream/handle/19716/299/PP%20339.pdf?sequence=1&isAllowed=y. [Accessed 11 December 2019].

[12] W. Kang and Y. Han, "A Simple and Realistic Pedestrian Model for Crowd Simulation and Application", 2012. Available: https://arxiv.org/pdf/1708.03080.pdf. [Accessed 12 December 2019].

[13] S. Sun, "Agent-based Crowd Simulation Modelling for a Gaming Environment", 2017. Available: https://scholar.uwindsor.ca/cgi/viewcontent.cgi?article=8401&context=etd. [Accessed 12 December 2019].

[14] *MASSIVE*. MASSIVE, 2001.

[15] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems", *Proceedings of the National Academy of Sciences*, vol. 99, no. 3, pp. 7280-7287, 2002. Available: 10.1073/pnas.082080899 [Accessed 12 December 2019].

[16] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics", *Physical Review E*, vol. 51, no. 5, pp. 4282-4286, 1995. Available: 10.1103/physreve.51.4282 [Accessed 12 December 2019].

[17] L. Torrey, "Crowd Simulation Via Multi-Agent Reinforcement Learning", 2010. Available: https://www.aaai.org/ocs/index.php/AIIDE/AIIDE10/paper/viewFile/2112/2550. [Accessed 12 December 2019].

[18] R. Sutton and A. Barton, "Reinforcement Learning: An Introduction", *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 1054-1054, 1998. Available: https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf. [Accessed 12 December 2019].

[19] T. Liu, Z. Liu, M. Ma, T. Chen, C. Liu and Y. Chai, "3D visual simulation of individual and crowd behavior in earthquake evacuation", *SIMULATION*, vol. 95, no. 1, pp. 65-81, 2018. Available: https://journals.sagepub.com/doi/full/10.1177/0037549717753294. [Accessed 15 December 2019].

[20] J. Funge, X. Tu and D. Terzopoulos, "Cognitive Modeling: Knowledge, Reasoning and Planning for Intelligent Characters", 2019. Available: http://www.dgp.toronto.edu/~funge/pdf/fttsig.pdf. [Accessed 15 December 2019].

[21] F. Durupınar, J. Allbeck, N. Pelechano and N. Badler, "Creating Crowd Variation with the OCEAN Personality Model", 2019. Available: https://www.cs.upc.edu/~npelechano/AAMAS08_Durupinar.pdf. [Accessed 15 December 2019].

[22] *Miarmy*. China: Basefount Company, 2011.

[23] *Golaem*. France: Golaem, 2011.

[24] *Oasys Mass Motion*. United Kingdom: Oasys, 2006.

[25] *uCrowds*. Netherlands.

[26] X. Shi, Z. Ye, N. Shiwakoti, D. Tang and J. Lin, "Examining effect of architectural adjustment on pedestrian crowd flow at bottleneck", Physica A: Statistical Mechanics and its Applications, vol. 522, pp. 350-364, 2019. Available: https://arxiv.org/ftp/arxiv/papers/1808/1808.07439.pdf. [Accessed 29 April 2020].

[27] "UDK | NavigationMeshReference", Docs UnrealEngine. [Online]. Available: https://docs.unrealengine.com/udk/Three/NavigationMeshReference.html. [Accessed: 25- Apr- 2020].

[28] C. Ferreira and M. Geig, "Get Started with the Unity* Entity Component System (ECS), C# Job System, and Burst Compiler", Intel Software Developer Zone, 2018. [Online]. Available: https://software.intel.com/en-us/articles/get-started-with-the-unity-entity-component-system-ecs-c-sharp-job-system-and-burst-compiler. [Accessed: 01- May- 2020].

[29] "ECS 1: Inheritance vs Composition and ECS Background ", Leatherbee, 2019. [Online]. Available: https://leatherbee.org/index.php/2019/09/12/ecs-1-inheritance-vs-composition-and-ecs-background/#Why_ECS_Its_Fast. [Accessed: 01- May- 2020].

[30] L. Meijer, "On DOTS: Entity Component System - Unity Technologies Blog", Unity Technologies Blog, 2019. [Online]. Available: https://blogs.unity3d.com/2019/03/08/on-dots-entity-component-system/. [Accessed: 01- May- 2020].

[31] "Physx", https://www.geforce.co.uk/. [Online]. Available: https://www.geforce.co.uk/hardware/technology/physx. [Accessed: 09- May- 2020].

[32] "Evacuation of premises - Kent Fire and Rescue Service", Kent.fire-uk.org. [Online]. Available: https://www.kent.fire-uk.org/your-safety/business-safety/top-tips-for-businesses/evacuation-of-premises/. [Accessed: 10- May- 2020].

[33] "Human Benchmark", Humanbenchmark.com. [Online]. Available: https://humanbenchmark.com/tests/reactiontime/statistics. [Accessed: 16- May- 2020].